La FAQ Debian GNU/Linux

Vous trouverez la liste des auteurs à Auteurs de la FAQ Debian

version 5.0.2, 2 Juin 2013

Résumé

Ce document répond aux questions fréquemment posées sur Debian GNU/Linux.

Copyright

Table des matières

Chapitre 1

Définitions et vue d'ensemble

1.1 Qu'est-ce que cette FAQ ?

Ce document présente les questions les plus fréquemment posées (ainsi que les réponses !) à propos de la distribution Debian (Debian GNU/Linux et les autres) et à propos du projet Debian. Des liens vers d'autres documentations sont indiqués le cas échéant : nous préférons ne pas citer de trop grands extraits des documentations externes à ce document. Vous verrez que certaines réponses supposent quelques connaissances des systèmes d'exploitation de type Unix. Nous essayons de supposer ces pré-requis les plus réduits possible : les réponses aux questions générales des débutants resteront simples.

Si vous ne trouvez pas réponse à votre question dans cette FAQ, vous devriez vérifier dans 'Quelles autres documentations existent sur le système Debian ?' page 53. Si cela ne vous aide pas, référez-vous à 'Commentaires' page 63.

1.2 Qu'est-ce que Debian GNU/Linux ?

Debian GNU/Linux est une *distribution* spécifique du système d'exploitation Linux disposant de nombreux paquets.

Debian GNU/Linux est :

— **complète** : actuellement, Debian inclut plus de 37400 logiciels. Les utilisateurs peuvent choisir quels paquets installer ; Debian fournit un outil à cette fin. Vous pouvez trouver une liste et la description des paquets actuellement disponibles dans Debian sur n'importe quel miroir (http://www.debian.org/distrib/ftplist) Debian.

— **libre d'utilisation et de distribution** : il n'y a aucune exigence d'adhésion ou de paiement à un établissement pour participer à sa distribution et à son développement. Tous les paquets qui font formellement partie de Debian GNU/Linux sont libres d'être redistribués, généralement sous les termes de la licence GNU GPL.

  Les archives FTP de Debian fournissent également environ 187 logiciels (dans les sections `non-free` et `contrib`), qui sont distribuables selon les conditions spécifiques incluses avec chaque paquet.

— **dynamique** : avec environ 990 volontaires qui contribuent constamment à la création et à l'amélioration du code, Debian évolue rapidement. Les archives FTP sont mises à jour deux fois par jour.

La plupart des utilisateurs de Linux font appel à une *distribution* spécifique de Linux, comme l'est Debian GNU/Linux. Cependant, les utilisateurs peuvent en principe obtenir le noyau Linux via Internet ou ailleurs et le compiler eux-mêmes. De la même manière, ils peuvent récupérer le code source de nombreuses applications, compiler les programmes et les installer sur leur système. Pour des programmes complexes, ce processus peut être non seulement long mais aussi source d'erreurs. Pour éviter cela, les utilisateurs choisissent souvent d'obtenir le système d'exploitation et l'ensemble des applications d'un des distributeurs de Linux. Ce qui distingue les différentes distributions Linux, ce sont les logiciels, les protocoles, et les pratiques employées pour empaqueter, installer et vérifier l'ensemble des applications sur les systèmes des utilisateurs, combinés avec des outils d'installation et d'administration, de la documentation, et d'autres services.

Debian GNU/Linux est le résultat d'un effort de volontaires pour créer un système d'exploitation, compatible Unix, libre et de grande qualité, complété d'un ensemble d'applications. L'idée d'un système libre de type Unix provient du projet GNU et plusieurs des applications qui rendent Debian GNU/Linux si utile ont été développées par le projet de GNU.

Pour Debian, le terme libre a le sens donné par le projet GNU (voir les principes du logiciel libre selon Debian (http://www.debian.org/social_contract#guidelines)). Quand nous parlons de logiciel libre, nous faisons référence à la liberté et non au prix (NdT : ambiguïté du mot anglais « free » signifiant libre et gratuit). Logiciel libre signifie que vous avez la liberté de distribuer des copies des logiciels, que vous recevez le code source ou que vous pouvez l'obtenir si vous le

voulez, que vous pouvez modifier le logiciel ou employer des parties du code dans de nouveaux projets libres et que vous savez que vous pouvez faire tout cela.

Le projet Debian a été créé par Ian Murdock en 1993, initialement sous le patronage du projet GNU de la Free Software Foundation. Aujourd'hui, les développeurs Debian le voient comme un descendant direct du projet GNU.

Comme Debian GNU/Linux est elle-même un logiciel libre, elle peut servir de base pour des distributions Linux à valeur ajoutée. En fournissant un système de base fiable et complet, Debian fournit aux utilisateurs Linux une compatibilité accrue, et permet à des créateurs de distribution Linux d'éliminer la duplication des efforts en se concentrant sur les choses qui rendent leur distribution particulière. Voir 'Je fais une distribution Linux particulière pour « un marché vertical ». Puis-je utiliser Debian GNU/Linux comme base du système et ajouter mes propres applications au-dessus ?' page 59 pour plus d'informations.

1.3 D'accord, maintenant je sais ce qu'est Debian, qu'est-ce que Linux ?

En résumé, Linux est le noyau d'un système d'exploitation de type Unix. Il a été à l'origine conçu pour les ordinateurs 386 (et plus récents) ; aujourd'hui Linux fonctionne également sur une douzaine d'autres systèmes. Linux est développé par Linus Torvalds et beaucoup d'informaticiens du monde entier.

En plus du noyau, un système Linux contient habituellement :

— un système de fichiers qui suit la norme de hiérarchie du système de fichiers Linux (« Filesystem Hierarchy Standard », FHS) http://www.pathname.com/fhs/,

— un ensemble d'utilitaires Unix, dont la plupart ont été développés par le projet GNU et la Free Software Foundation.

La combinaison du noyau Linux, du système de fichiers, des utilitaires GNU de la FSF, et des autres utilitaires a été conçue pour être en conformité avec la norme POSIX (IEEE 1003.1). Voir 'Comment le code source compatible Debian l'est-il avec d'autres systèmes Unix ?' page 14.

Pour plus d'informations sur Linux, voir le document en anglais What is Linux (http://www.linux.org/info/) par Linux Online (http://www.linux.org/).

1.4 Debian fournit uniquement GNU/Linux ?

Actuellement, Debian n'est disponible que pour Linux, mais avec Debian GNU/Hurd et Debian pour les noyaux BSD, nous avons aussi commencé à fournir comme plate-forme de développement des systèmes d'exploitations autres que ceux basés sur Linux. Cependant, ces portages non Linux ne sont pas encore officiellement publiés.

Debian GNU/Hurd fut le premier effort de portage.

Le Hurd est un ensemble de serveurs s'exécutant au dessus du micronoyau GNU Mach. Ensemble, ils forment la base du système d'exploitation GNU.

Veuillez vous reporter au site http://www.gnu.org/software/hurd/ pour plus d'informations en général sur GNU/Hurd et au site http://www.debian.org/ports/hurd/ pour des informations particulières à Debian GNU/Hurd.

Le deuxième effort de portage est celui du noyau BSD. Les développeurs travaillent à la fois sur les noyaux NetBSD et les noyaux FreeBSD.

Veuillez consulter http://www.debian.org/ports/#nonlinux pour plus d'informations sur les portages autres que Linux.

1.5 Quelles sont les différences entre Debian GNU/Linux et les autres distributions Linux ? Pourquoi choisir Debian plutôt qu'une autre ?

Ces fonctions principales distinguent Debian des autres distributions Linux :

Liberté : Comme déclaré dans le Contrat social de Debian (http://www.debian.org/social_contract), Debian restera 100% libre. Le projet Debian est très strict quant à fournir des logiciels vraiment libres. Les principes appliqués pour déterminer si un travail est « libre » sont fournis dans Les principes du logiciel libre selon Debian (http://www.debian.org/social_contract#guidelines).

Le système de gestion de paquets de Debian : Le système entier ou n'importe quel composant individuel peut être mis à jour sans reformater, sans perdre les fichiers de configuration personnalisés et (dans la plupart des cas) sans redémarrer le système. Aujourd'hui, la plupart des distributions Linux disponibles ont leur propre système de gestion de paquets ; le système de gestion de paquets de Debian est unique et particulièrement robuste. (voir 'Les bases du système de gestion des paquets Debian' page 27)

Développement ouvert : Alors que d'autres distributions Linux sont développées par des individus, des petits groupes fermés, ou des fournisseurs commerciaux. Debian est la seule distribution Linux majeure qui est développée coopérativement par beaucoup d'individus grâce à Internet, dans le même esprit que Linux et d'autres logiciels libres.

Plus de 990 responsables de paquet travaillent bénévolement sur plus de 37400 paquets et participent à l'amélioration de Debian GNU/Linux. Les développeurs Debian contribuent au projet non pas en programmant de nouvelles applications (dans la plupart des cas) mais en empaquetant les logiciels existants suivant les normes du projet, en envoyant des rapports de bogue aux développeurs amont et en fournissant de l'aide aux utilisateurs. Voir aussi les informations sur comment devenir un contributeur dans 'Comment devenir un développeur Debian ?' page 57.

Le système d'exploitation universel : Debian fournit plus de 37400 paquets (`http://packages.debian.org/stable/`) et fonctionne sur 10 architectures (`http://www.debian.org/ports/`). C'est bien plus que dans n'importe quelle autre distribution GNU/Linux. Consultez 'Quelles sont les applications et les logiciels de développement disponibles sous Debian GNU/Linux ?' page 17 pour avoir une vue d'ensemble des logiciels proposés et 'Sur quelle architecture matérielle fonctionne Debian GNU/Linux ?' page 13 pour une description des plateformes matérielles prises en charge.

Le système de gestion des bogues : La répartition géographique des développeurs Debian demande des outils sophistiqués et une communication rapide des bogues et de leur résolution pour accélérer le développement du système. Les utilisateurs sont encouragés à envoyer des rapports de bogue dans un modèle formel, qui est rapidement accessible par les archives web ou par courrier électronique. Reportez-vous aux informations supplémentaires dans cette FAQ, sur la gestion des bogues dans 'Y a-t-il une base des bogues connus ?' page 55.

La charte Debian : Debian a des spécifications étendues des normes de qualité, la charte Debian. Ce document définit les exigences de qualité et les normes que doit satisfaire chaque paquet Debian.

Pour d'autres informations sur le sujet, veuillez voir la page web sur les raisons pour choisir Debian (`http://www.debian.org/intro/why_debian`).

1.6 Comment le projet Debian s'insère-t-il avec le projet GNU de la Free Software Foundation ?

Le système Debian est construit sur les idéaux du logiciel libre d'abord soutenus par la Free Software Foundation (`http://www.gnu.org/`) et en particulier par Richard Stallman (`http://www.stallman.org/`). Les outils de développement puissants ainsi que les utilitaires, et les applications de la FSF sont également une partie principale du système Debian.

Le projet Debian est indépendant de la FSF, cependant nous communiquons régulièrement et coopérons sur différents projets. La FSF a explicitement demandé que nous appelions notre système « Debian GNU/Linux » et nous sommes heureux de nous conformer à cette demande.

L'objectif de longue date de la FSF est de développer un nouveau système d'exploitation appelé GNU, basé sur Hurd (`http://www.gnu.org/software/hurd/`). Debian travaille avec la FSF sur ce système, nommé Debian GNU/Hurd (`http://www.debian.org/ports/hurd/`).

1.7 Comment prononce-t-on Debian et quel est le sens de ce mot ?

Le nom du projet est prononcé Deb'-i-an, avec un e court dans Deb et l'accentuation portée sur la première syllabe. Ce mot est la contraction des noms Debra et Ian Murdock, qui a fondé le projet. (Les dictionnaires semblent laisser une certaine ambiguïté dans la prononciation de Ian, mais Ian préfère i'-an.)

Chapitre 2

Obtenir et installer Debian GNU/Linux

La documentation officielle décrivant les instructions d'installation est le Guide d'installation Debian GNU/Linux (http://www.debian.org/releases/stable/installmanual). Quelques informations supplémentaires sur le téléchargement et l'installation de Debian GNU/Linux seront mentionnées ici.

2.1 Quelle est la dernière version de Debian ?

Actuellement il existe trois versions de Debian GNU/Linux :

La version 7.0, c.-à-d. la distribution « stable » ou wheezy, c'est la version stable et bien testée, seuls sont incorporés les changements dus à des problèmes importants de sécurité ou d'utilisation.

La distribution « testing », actuellement appelée jessie, c'est dans cette distribution que sont placés les paquets qui seront dans la prochaine distribution « stable » ; ils ont déjà été testés lors de leur passage dans « unstable » mais ne sont pas encore complètement prêts. Cette distribution est mise à jour plus souvent que la distribution « stable », mais moins souvent que la distribution « unstable ».

La distribution « unstable », c'est la version actuellement en développement ; elle est continuellement mise à jour. Vous pouvez récupérer des paquets archives instables sur n'importe quel site FTP de Debian et les employer pour améliorer votre système, mais vous ne devez pas vous attendre à ce que le système soit aussi utilisable ou aussi stable qu'avant. C'est pourquoi cette version s'appelle « unstable ».

Reportez-vous à la question 'Combien de distributions de Debian y a-t-il ?' page 21 pour plus d'informations.

2.2 Existe-t-il des mises à jour de la distribution « stable » ?

Aucune nouvelle fonctionnalité n'est ajoutée à la version stable. Une fois qu'une version de Debian est publiée et considérée comme stable, les seules mises à jour de la distribution sont des mises à jour de sécurité. C'est-à-dire que seuls les paquets pour lesquels une faille de sécurité a été trouvée après la publication sont mis à jour. Toutes les mises à jour de sécurité sont effectuées à partir du serveur security.debian.org (ftp://security.debian.org).

Les mises à jour de sécurité ont un but : fournir une correction pour une faille de sécurité. Elles ne sont pas là pour ajouter discrètement des modifications dans la version stable sans passer par la procédure normale de version intermédiaire. En conséquence, les correctifs pour les paquets avec des problèmes de sécurité ne sont pas une mise à jour du logiciel, mais un rétroportage par l'équipe de sécurité du correctif nécessaire pour la version distribuée dans stable.

Pour plus d'informations liées à la gestion de la sécurité, veuillez lire la FAQ de l'équipe Debian sur la sécurité (http://www.debian.org/security/faq) ou le Manuel de sécurisation de Debian (http://www.debian.org/doc/manuals/securing-debian-howto/).

2.3 Où et comment puis-je obtenir les disques d'installation de Debian ?

Vous pouvez obtenir les disques d'installation en téléchargeant les fichiers appropriés depuis les miroirs Debian (http://www.debian.org/mirror/list).

Consultez la page Debian GNU/Linux sur CD (http://www.debian.org/CD) pour plus d'informations sur les images CD et DVD.

2.4 Comment installer Debian depuis les CD-ROM ?

Installer Debian depuis un CD est tout simple : configurez votre système pour démarrer depuis le CD, insérez le CD et redémarrez. Votre système démarrera l'installateur Debian. Consultez le Manuel d'installation pour la distribution Debian GNU/Linux (http://www.debian.org/releases/stable/installmanual) pour plus d'informations.

2.5 Pourquoi le CD-ROM officiel de la publication stable contient-il des liens symboliques pour *frozen* et *unstable* ? Je pensais que ce CD contenait seulement *stable* !

Les images officielles des CD Debian contiennent en effet des liens symboliques comme :

```
/dists/frozen -> wheezy/
/dists/stable -> wheezy/
/dists/testing -> wheezy/
/dists/unstable -> wheezy/
```

afin qu'ils puissent fonctionner dans le cas où votre sources.list possède une entrée du type

```
deb cdrom:[<nom du CD>]/ unstable main [...]
```

.

La présence de ces liens symboliques *ne* signifie *pas* que cette image est celle de la distribution *unstable*, *testing* ou encore autre chose. Reportez-vous au nom du CD présent dans `/.disk/info` pour savoir quelle version de Debian ce CD contient. Cette information est également présente dans le fichier `/README.txt` sur le CD.

Consultez http://www.debian.org/releases/ pour savoir quelles sont les distributions *stable* et *testing* actuelles.

2.6 Puis-je installer directement depuis un site Internet distant ?

Oui. Vous pouvez démarrer le système d'installation de Debian depuis un ensemble de fichiers que vous pouvez télécharger de notre site ftp ou de l'un de ses miroirs.

Vous pouvez télécharger une petite image de CD, créer avec elle un CD amorçable, installer le système de base à partir de l'image puis finir l'installation à travers le réseau. Pour plus d'informations, veuillez-vous reporter à http://www.debian.org/CD/netinst/.

Vous pouvez aussi télécharger les fichiers d'image de disquette encore plus petits, créer les disquettes amorçables à partir d'elles, commencer la procédure d'installation et obtenir le reste de Debian depuis le réseau.

2.7 Existe-t-il d'autres alternatives pour démarrer l'installateur ?

Oui. Mis à part les CD ou DVD, vous pouvez installer Debian en démarrant depuis des disquettes, une clé USB, directement depuis le disque dur, ou encore par amorçage réseau et TFTP. Il est possible d'effectuer de manière complètement automatique les installations sur plusieurs ordinateurs. N.B. : toutes les méthodes ne sont pas possibles sur toutes les architectures. Une fois l'installateur démarré, le reste du système peut être téléchargé par le réseau ou depuis un média local. Consultez le Manuel d'installation pour la distribution Debian GNU/Linux (http://www.debian.org/releases/stable/installmanual) pour plus d'informations.

Chapitre 3

Choisir une distribution de Debian

Il existe plusieurs versions de Debian. Choisir la version appropriée est une décision importante. Cette section donne des informations utiles aux utilisateurs désirant choisir la version qui convient le mieux à leurs besoins, et répond également aux questions qui peuvent être soulevées lors de ce choix. Il ne s'agit pas de répondre à la question « pourquoi vous devriez choisir Debian ? » mais plutôt « quelle version de Debian ? ».

Pour plus d'informations sur les versions disponibles, consultez 'Combien de distributions de Debian y a-t-il ?' page 21.

3.1 Quelle version de Debian (stable/testing/unstable) me convient le mieux ?

La réponse est un peu compliquée. Cela dépend de ce que vous voulez faire. Une solution serait de demander à un ami qui utilise Debian. Mais cela signifierait que vous ne pouvez pas prendre votre décision de manière indépendante. Vous devriez pouvoir prendre votre décision après la lecture de ce chapitre.

— Si la sécurité ou la stabilité vous importent par-dessus tout : installez la version stable. Il s'agit du choix favori des utilisateurs.

— Si vous êtes un nouvel utilisateur installant une machine de bureau, démarrez avec la version stable. Certains logiciels sont un peu anciens, mais c'est l'environnement le moins boggué pour travailler. Vous pourrez basculer vers la version *unstable* plus récente une fois que vous serez plus en confiance.

— Si vous êtes un utilisateur d'une machine de bureau avec de l'expérience dans les systèmes Linux, et que cela nous vous dérange pas de faire face à des bugs étranges de temps en temps, utilisez *unstable*. Cette version comporte tous les derniers logiciels, et les bogues sont généralement corrigés rapidement.

— Si vous utilisez un serveur, en particulier avec des contraintes importantes de stabilité ou s'il est connecté à Internet, installez la version *stable*. Il s'agit de loin du choix le plus sûr et le plus solide.

Les questions suivantes vous donnent, nous l'espérons, plus de détails sur ces choix. Si après avoir lu toute cette FAQ, vous n'arrivez pas à choisir, restez sur le choix de la distribution stable.

3.1.1 Vous m'avez demandé d'installer stable, mais en stable tel et tel matériel ne sont pas détectés ou ne fonctionnent pas. Que dois-je faire ?

Essayez de rechercher sur le web à l'aide d'un moteur de recherche afin de voir si quelqu'un d'autre a été capable de faire fonctionner ce matériel avec *stable*. La plupart du matériel devrait fonctionner correctement avec *stable*. Dans le cas où vous avez du matériel à la pointe, il peut ne pas fonctionner avec *stable*. Dans ce cas, vous pourriez installer ou basculer vers *unstable*.

Concernant les ordinateurs portables, `http://www.linux-on-laptops.com/` est un très bon site web permettant de savoir si quelqu'un d'autre a fait fonctionner un modèle précis sous Linux. Ce site web n'est pas spécifique à Debian, mais constitue néanmoins une source fantastique. À notre connaissance, il n'existe pas de tel site pour les ordinateurs de bureau.

Une autre option est de demander sur la liste de diffusion debian-user en envoyant un courriel à debian-user@lists.debian.org. Il est possible d'envoyer des messages à liste sans être abonné. Les archives sont disponibles depuis `http://lists.debian.org/debian-user/`, où se trouvent également les informations sur l'abonnement. Nous vous incitons fortement à poser vos questions sur la liste de diffusion plutôt que sur irc (`http://www.debian.org/support`). En effet, les messages de la liste de diffusion sont archivés, ce qui peut aider les personnes ayant le même problème que le vôtre à trouver la solution.

3.1.2 Y aura-t-il des différences de versions des paquets entre les différentes distributions ?

Oui. La distribution *unstable* inclut les versions les plus récentes, mais qui ne sont pas encore suffisamment testées et peuvent donc comporter des bogues.

En revanche, *stable* contient d'anciennes versions des paquets, qui sont ainsi bien éprouvés et sont moins susceptibles de comporter des bogues.

Les paquets dans *testing* se trouvent entre ces deux extrêmes.

3.1.3 La distribution stable contient des paquets vraiment obsolètes. Regardez Kde, Gnome, Xorg ou même le noyau. Ils sont très anciens. Pourquoi cette situation ?

Il est possible que vous ayez raison. L'âge des paquets dans *stable* dépend de la date de la publication. Comme il s'écoule généralement plus d'un an entre les publications, vous pouvez penser que stable contient des vieilles versions des paquets. Cependant, ils ont été testés intensivement. On peut raisonnablement dire que ces paquets ne comportent aucun bogue grave connu, ni de trous de sécurité, etc. Les paquets de *stable* s'intègrent de manière transparente avec les autres paquets de *stable*. Ces caractéristiques sont très importantes pour les serveurs de production qui doivent fonctionner 24 heures par jour et 7 jours par semaine.

En revanche, les paquets de *testing* ou *unstable* peuvent comporter des bogues cachés, des trous de sécurité, etc. De plus, certains de ces paquets peuvent ne pas fonctionner comme prévu. Généralement les personnes travaillant sur un unique ordinateur de bureau préfèrent avoir les paquets les plus récents. La distribution *unstable* est la solution pour cette catégorie d'utilisateurs.

Comme vous pouvez le constater, la stabilité et la nouveauté sont les deux extrémités du spectre. Besoin de stabilité ? Utilisez la distribution *stable*. Si vous voulez travailler avec les paquets les plus récents, alors installez la distribution *unstable*.

3.1.4 Et si je désire changer de distribution, puis-je le faire ?

Oui, mais il s'agit d'une opération irréversible. Vous pouvez mettre à jour votre système de stable –> testing –> unstable. L'opération en sens inverse est impossible. Soyez bien sûr de votre décision si vous prévoyez d'installer ou de mettre à jour pour *unstable*.

En fait, si vous êtes un expert, prêt à passer du temps, vraiment prudent et que vous savez ce que vous faites, alors il peut être possible de passer de la distribution *unstable* vers *testing* puis vers *stable*. Les scripts d'installation des paquets ne sont pas prévus pour cela, il est donc possible par exemple que vous perdiez certains fichiers de configuration au cours de l'opération.

3.1.5 Faut-il installer testing ou unstable ?

C'est plutôt subjectif. Il n'existe pas de réponse parfaite mais seulement une « estimation sage » à faire lors du choix entre unstable et testing. L'auteur conseille dans l'ordre de préférence : *stable*, *unstable* puis *testing*. Le problème est le suivant :

- Stable est solide comme un roc. Elle est incassable.
- Testing est cassée moins souvent que *unstable*. Mais lorsque cela arrive, la correction met du temps à être appliquée. Des fois il peut s'agir de plusieurs jours, et dans certains cas plusieurs mois.
- Unstable change beaucoup, et peut être cassée à n'importe quel moment. Cependant, les problèmes sont souvent corrigés en quelques jours et cette distribution offre toujours les dernières versions des logiciels empaquetés pour Debian.

Pourtant il existe des cas où utiliser testing serait plus avantageux qu'unstable. L'auteur a rencontré ce cas lors de la transition de gcc3 à gcc4. Le paquet `labplot` était impossible à installer sur une machine unstable car certaines de ses dépendances avaient passé la transition gcc4 et d'autres pas. Au même moment, le paquet de testing était installable sur une machine testing puisque les paquets ayant effectué la transition gcc4 n'avaient pas atteint testing.

3.1.6 Vous parlez de cas où testing est cassée. Qu'entendez-vous par là ?

Parfois, un paquet peut ne pas être installable par les outils de gestion des paquets. Parfois, un paquet peut ne pas être disponible du tout, éventuellement supprimé temporairement en raison de bogues ou de dépendances non résolues. Enfin, un paquet peut s'installer mais ne pas offrir un comportement satisfaisant.

Lorsque cela arrive, on dit que la distribution est cassée (du moins pour ce paquet).

3.1.7 Pourquoi testing peut-elle être cassée pendant plusieurs mois ? Les correctifs introduits dans unstable n'arrivent-ils par directement dans testing ?

Les corrections de bogues et les améliorations introduites dans la distribution *unstable* atterrissent dans *testing* après un certain nombre de jours. Disons que ce seuil est de 10 jours. Les paquets d'unstable arrivent dans testing seulement si aucun bogue critique pour la publication (« RC » pour « Release Critical ») n'est signalé à leur égard. Si un bogue RC est signalé sur un paquet d'unstable, il n'entrera pas dans testing avant les 10 prochains jours.

En effet on considère que si le paquet a un problème, celui-ci sera découvert par les utilisateurs d'unstable et sera corrigé avant que le paquet puisse atteindre testing. Cela permet à testing de rester utilisable la plupart du temps. Le concept est génial la plupart du temps, mais les choses ne sont jamais si simples. Considérez par exemple la situation suivante :

— Vous êtes intéressé par le paquet XYZ.
— Le 10 juin, la version dans testing est XYZ-3.6 et dans unstable XYZ-3.7
— 10 jours après, XYZ-3.7 migre d'unstable vers testing.
— Donc le 20 juin, testing et unstable ont toutes deux XYZ-3.7.
— Un utilisateur de la distribution testing repère qu'une nouvelle version de XYZ est disponible et met à jour sa version XYZ-3.6 en XYZ-3.7.
— Le 25 juin, quelqu'un utilisant testing ou unstable découvre un bogue RC dans XYZ-3.7 et le signale dans le BTS.
— Le responsable de XYZ corrige ce bogue et l'envoie vers unstable le 30 juin. On suppose ici que 5 jours sont nécessaires au responsable pour corriger et envoyer la nouvelle version. Ce chiffre de 5 ne doit pas être pris au pied de la lettre ; il peut être supérieur ou inférieur, suivant la sévérité du bogue RC.
— Cette nouvelle version arrivée dans unstable, XYZ-3.8 est programmée pour atteindre testing le 10 juillet.
— Mais le 5 juillet, un autre utilisateur découvre un autre bogue dans XYZ-3.8.
— Considérons que le responsable de XYZ corrige ce nouveau bogue et envoie la nouvelle version en 5 jours.
— Ainsi le 10 juillet, testing propose XYZ-3.7 alors que unstable propose XYZ-3.9.
— Cette nouvelle version XYZ-3.9 est reprogrammée pour atteindre testing le 20 juillet.
— Puisque vous utilisez testing et que XYZ-3.7 est boggué, vous ne pourrez sans doute utiliser XYZ qu'après le 20 juillet. Ainsi vous aurez passé près d'un mois avec une version XYZ cassée.

La situation peut être bien plus compliquée, si par exemple XYZ dépend de 4 autres paquets. Cela peut rendre la distribution testing inutilisable pendant plusieurs mois. Le scénario ci-dessus est fictif et peut se produire dans la réalité, même si de tels cas sont rares.

3.1.8 Du point de vue d'un administrateur, quelle distribution requiert le plus d'attention ?

L'une des principales raisons pour laquelle de nombreuses personnes choisissent Debian plutôt que d'autres distributions, est qu'elle demande très peu d'administration. Les gens désirent un système qui se contente de fonctionner. De manière générale, on peut dire que la distribution *stable* exige très peu de maintenance alors que *testing* et *unstable* demandent une maintenance constante à l'administrateur. Si vous utilisez stable, tout ce dont vous avez à vous soucier est d'appliquer les mises à jour de sécurité. Si vous utilisez testing ou unstable, c'est une bonne idée d'être au courant des nouveaux bogues découverts dans les paquets installés, les correctifs et fonctionnalités introduits, etc.

3.1.9 Que se passe-t-il lors d'une nouvelle publication ?

Cette question ne vous aidera pas à choisir une distribution de Debian, mais vous vous la poserez tôt ou tard.

La distribution stable est actuellement wheezy ; la prochaine distribution stable s'appelera jessie. Considérons le cas particulier de ce qui se produira lors de la publication de jessie comme nouvelle version stable.

— oldstable = squeeze ; stable = wheezy ; testing = jessie ; unstable = sid
— Unstable fait toujours référence à sid, indépendamment du processus de publication.
— Les paquets migrent continuellement de sid vers testing (c.-à.-d. jessie). Mais les paquets dans stable (c.-à-d. wheezy) restent les mêmes sauf s'il existe des mises à jour de sécurité.
— Après un certain temps *testing* est gelée, mais reste appelée *testing*. À partir de ce moment-là, plus aucun nouveau paquet d'unstable ne peut migrer vers testing, à moins qu'ils corrigent des bogues critiques pour la publication (RC).

- Lors du gel de testing, tous les correctifs introduits doivent être vérifiés par les membres de l'équipe de publication. Cela permet de s'assurer qu'aucun problème sévère inconnu n'est introduit dans la distribution *testing* gelée.
- Les bogues RC de la testing gelée sont réduits à zéro.
- La testing gelée sans bogue RC sera publiée comme nouvelle version stable. Dans notre exemple, cette nouvelle publication stable sera appelée jessie.
- À cette étape, oldstable = wheezy, stable = jessie. Les contenus de *stable* et la testing gelée sont alors identiques.
- Une nouvelle testing est dérivée de l'actuelle version de développement (unstable).
- Les paquets commencent de nouveau à migrer de sid vers testing. La communauté Debian travaille déjà à préparer la nouvelle publication stable.

3.1.10 Je dispose d'une machine de bureau / de serveurs utilisant Debian. Comment savoir quelle distribution j'utilise ?

Dans la plupart des cas, c'est très simple à savoir. Regardez le contenu du fichier `/etc/apt/sources.list`. Vous trouverez une entrée similaire à la ligne suivante :

```
deb http://ftp.us.debian.org/debian/ unstable main contrib
```

Le troisième champ (« unstable » dans l'exemple ci-dessus) indique quelle distribution le système utilise actuellement.

Vous pouvez également utiliser `lsb_release` (disponible dans le paquet `lsb-release`). Si vous exécutez ce programme sur un système utilisant *unstable*, vous obtiendrez :

```
$ lsb_release  -a
LSB Version:    core-2.0-noarch:core-3.0-noarch:core-3.1-noarch:core-2.0-ia32:core-3.0-ia32:core-3.1-ia32
Distributor ID: Debian
Description:    Debian GNU/Linux unstable (sid)
Release:    unstable
Codename:   sid
```

Pourtant, cela n'est pas toujours si facile. Certains systèmes peuvent comporter un fichier `sources.list` avec plusieurs entrées vers différentes distributions. Cela peut arriver lorsque l'administrateur utilise des paquets depuis différentes distributions. Ce mélange de plusieurs distributions est appelé épinglage (apt-pinning).

3.1.11 J'utilise actuellement stable. Puis-je changer pour testing ou unstable ? Si oui, comment ?

Si vous utilisez actuellement stable, alors le troisième champ du fichier `/etc/apt/sources.list` sera soit wheezy ou stable. Vous devez remplacer cette valeur par le nom de la version que vous voulez utiliser. Si vous voulez utiliser testing, alors remplacez ce troisième champ de `/etc/apt/sources.list` par testing. Si vous souhaitez utiliser unstable, alors remplacez le troisième champ par unstable.

Actuellement testing est appelée jessie. Ainsi, si vous remplacez le troisième champ de `/etc/apt/sources.list` en jessie, vous utiliserez alors testing. Mais lorsque jessie deviendra stable, vous continuerez à utiliser jessie.

Unstable est toujours appelée Sid. Si vous remplacez le troisième champ de `/etc/apt/sources.list` par sid, vous utiliserez alors unstable.

Actuellement, Debian propose des mises à jour de sécurité pour testing mais pas pour unstable, puisque les correctifs pour unstable sont directement appliqués dans l'archive standard. Ainsi si vous utilisez unstable, assurez-vous d'avoir supprimé dans `/etc/apt/sources.list` les lignes relatives aux mises à jour de sécurité.

Si des notes de publication sont disponibles pour la distribution vers laquelle vous effectuez la mise à niveau (même si la distribution n'a pas été encore publiée), il serait sage de les consulter car elles peuvent fournir des informations expliquant comment vous devriez procéder à la mise à niveau.

Néanmoins, une fois les changements ci-dessus effectués, vous pouvez exécuter `aptitudeupdate` puis installer les paquets de votre choix. Notez qu'installer un paquet d'une autre distribution peut déclencher la mise à jour de la moitié de votre système. Si vous installez des paquets individuels, vous finirez avec un système utilisant un mélange de plusieurs distributions.

Dans certaines situations, il préférable d'effectuer la mise à niveau complète vers la nouvelle distribution en exécutant `apt-get dist-upgrade`, `aptitude safe-upgrade` ou `aptitude full-upgrade`. Lisez les pages de manuel d'apt et d'aptitude pour plus d'informations.

3.1.12 J'utilise actuellement testing (jessie). Que se passera-t-il lors de la prochaine publication ? Utiliserai-je encore testing ou ma machine utilisera la nouvelle distribution stable ?

Cela dépend des entrées du fichier `/etc/apt/sources.list`. Si vous utilisez actuellement testing, les entrées sont similaires soit à :

```
deb http://ftp.us.debian.org/debian/ testing main
```

soit à :

```
deb http://ftp.us.debian.org/debian/ jessie main
```

Si le troisième champ de `/etc/apt/sources.list` a pour valeur « testing », alors vous utiliserez testing même après la prochaine publication. Ainsi après la publication de jessie, vous utiliserez une nouvelle distribution de Debian dotée d'un nouveau nom de code. Les changements peuvent ne pas être visibles au départ mais deviendront évidents dès que les nouveaux paquets d'unstable atteindront la distribution testing.

En revanche, si le troisième champ contient « jessie », alors vous utiliserez la distribution stable (puisque jessie sera alors la nouvelle distribution stable).

3.1.13 Cela reste confus pour moi. Que devrais-je installer selon vous ?

Si vous n'êtes pas certain, le meilleur pari est la distribution stable.

3.2 Et à propos de Knoppix, Linex, Ubuntu, et des autres ?

Ces distributions ne viennent pas du projet Debian ; elles sont *basées sur Debian*. Bien qu'il y ait de nombreuses similarités et de points communs, il existe également des différences cruciales.

Toutes ces distributions ont leurs propres avantages et sont appropriées à certains utilisateurs spécifiques. Pour plus d'informations, consultez la page du site Debian sur les distributions basées sur Debian (`http://www.debian.org/misc/children-distros`).

3.2.1 Je sais que Knoppix/Linex/Ubuntu/. . . est basée sur Debian. Donc après l'avoir installée sur disque dur, puis-je utiliser l'outil apt de gestion des paquets ?

Ces distributions sont basées sur Debian, mais elles ne sont pas Debian. Il vous sera toujours possible d'utiliser l'outil apt de gestion des paquets en pointant le fichier `/etc/apt/sources.list` vers les dépôts de ces distributions. Mais dans ce cas vous n'utilisez pas Debian, mais une autre distribution. Elles ne sont pas identiques.

Dans la plupart des situation si vous utilisez une distribution particulière, vous devriez n'utiliser que celle-ci et non pas la mélanger avec des paquets d'autres distributions. De nombreux dysfonctionnements apparaissent lorsque des personnes essaient d'installer des paquets Debian alors qu'ils utilisent une autre distribution. Le fait qu'elles utilisent le même format et nom (.deb) ne les rend pas nécessairement compatibles.

Par exemple, Knoppix est une distribution Linux conçue pour être démarrée sur un live CD alors que Debian est conçue pour être installée sur disque dur. Knoppix est géniale lorsque vous voulez savoir si un matériel particulier fonctionne, ou si vous voulez expérimenter comment est un système Linux, etc. Knoppix est adaptée pour une démonstration tandis que Debian est conçu pour fonctionner 24/7. De plus, le nombre de paquets disponibles et le nombre d'architectures prises en charge par Debian sont largement plus importants que ceux de Knoppix.

Si vous voulez utiliser Debian, le mieux est d'installer directement Debian après l'avoir téléchargé. Même s'il est possible d'installer Debian en passant par d'autres distributions, comme Knoppix, cette méthode nécessite de très bonnes connaissances du système. Si vous lisez cette FAQ, nous supposons que Debian et Knoppix sont nouveaux pour vous. Dans ce cas, économisez vous beaucoup de soucis plus tard et installez Debian dès le départ.

3.2.2 J'ai installé Knoppix/Linex/Ubuntu/... sur mon disque dur. Maintenant j'ai un problème. Que devrais-je faire ?

Il vous est conseillé de ne pas utiliser les forums Debian (ni les listes de diffusion ni IRC) pour demander de l'aide, car les gens qui pourraient vous conseiller penseront que vous faites fonctionner un système Debian, et les « correctifs » fournis pourraient ne pas convenir à votre système. Ils pourraient même aggraver le problème auquel vous êtes confronté.

Utilisez les forums de la distribution spécifique que vous utilisez en premier. Si vous n'obtenez vraiment pas d'aide ou que l'aide fournie ne corrige pas votre problème, vous pourriez vouloir essayer de demander sur les forums Debian mais gardez à l'esprit les conseils du paragraphe précédent.

3.2.3 J'utilise Knoppix/Linex/Ubuntu/... et maintenant je veux utiliser Debian. Comment puis-je migrer ?

Considérez le changement d'une distribution basée sur Debian vers Debian juste comme un changement d'un système d'exploitation vers un autre. Vous devriez effectuer une sauvegarde de toutes vos données et réinstaller le système d'exploitation de zéro. Vous ne devriez pas tenter de « mettre à jour » vers Debian en utilisant les outils de gestion de paquets, car vous pourriez finir avec un système inutilisable.

Si toutes vos données d'utilisateur (i.e. votre `/home`) sont dans une partition séparée, migrer vers Debian est en effet assez simple ; vous avez juste à dire au système d'installation de monter (mais de ne pas reformater) cette partition lors de la réinstallation. Effectuer des sauvegardes de vos données aussi bien que de la configuration de votre système précédent (i.e. `/etc/` et, peut-être, `/var/`) est toujours conseillé.

Chapitre 4

Problèmes de compatibilité

4.1 Sur quelle architecture matérielle fonctionne Debian GNU/Linux ?

Debian GNU/Linux fournit le code source complet pour tous les programmes, donc elle peut fonctionner sur tous les systèmes supportant le noyau Linux. Voir la FAQ Linux (http://en.tldp.org/FAQ/Linux-FAQ/intro.html#DOES-LINUX-RUN-ON-MY-COMPUTER) pour plus de détails.

La version actuelle de Debian GNU/Linux, la 7.0, contient une distribution binaire complète pour les architectures suivantes :

— *amd64* : ceci couvre les systèmes avec des processeurs AMD 64 bits avec l'extension AMD64, ainsi que tous les processeurs Intel avec l'extension EM64T, et fournit un environnement utilisateur en 64 bits.
— *armel* : les machines ARM petit-boutistes (« little-endian »).
— *armhf* : une alternative à l'architecture armel, destinée aux machines ARMv7 avec unité de calcul flottant.
— *i386* : ceci couvre les systèmes basés sur les processeurs Intel et compatibles, incluant les processeurs Intel 386, 486, Pentium, Pentium Pro, Pentium II (Klamath et Celeron), Pentium III, et la plupart des processeurs compatibles de AMD, Cyrix et autres.
— *ia64* : les ordinateurs Intel IA-64 (« Itanium »).
— *mips* : les systèmes MIPS gros-boutiens de SGI, Indy et Indigo2 ; *mipsel* : les machines MIPS petit-boutiennes, Digital DECstations.
— *powerpc* : ceci couvre certaines machines IBM/Motorola PowerPC, incluant les modèles Apple Macintosh PowerMac, et les architectures libres CHRP et PReP.
— *s390* : les systèmes mainframe S/390 de IBM.
— *s390x* : un portage 64 bits pour les machines IBM System z, destiné à remplacer s390.
— *sparc* : ceci couvre les SPARC de SUN et la plupart des systèmes UltraSPARC, ainsi que certains de leurs remplaçants au sein de l'architecture sun4.

Le développement de nouvelles distributions binaires est en cours pour : *armhf* (systèmes embarqués ARM avec unité de calcul flottant), *arv32* (architecture RISC 32 bits d'Atmel), *m32* (architecture RISC 32 bits de Renesas Technology), *s390x* (environnement 64 bits pour les systèmes IBM System Z) et *sh* (processeurs Hitachi SuperH).

La prise en charge de l'architecture *m68k* a été arrêtée depuis Etch (Debian 4.0), car elle ne remplissait plus les critères fixés par les responsables de publication de Debian. Cette architecture couvre les Amiga et les ATARI avec un processeur 680x0 avec x supérieur à 2 ; avec MMU. Cependant, ce portage est toujours actif et disponible à l'installation même s'il ne fait pas partie de la publication stable officielle. Ce portage pourrait être réactivé lors de futures publications.

La prise en charge des architectures *hppa* (systèmes PA-RISC de Hewlett-Packard) et *alpha* (systèmes Alpha de Compaq/Digital) a été arrêtée depuis Squeeze (Debian 6.0) pour des raisons similaires. L'architecture *arm* a été remplacée par *armel*.

Pour plus d'informations sur les portages disponibles, consultez les pages web sur les portages (http://www.debian.org/ports/).

Pour de plus amples informations sur le démarrage, le partitionnement du disque, l'activation des périphériques PCMCIA (« PC Card ») et sur des questions semblables, veuillez-vous référer aux instructions du manuel d'installation, qui est disponible sur notre site web à l'adresse http://www.debian.org/releases/stable/installmanual.

4.2 Quels noyaux utilise Debian GNU/Linux ?

Debian contient une distribution binaire complète pour les noyaux suivants :

— Le noyau FreeBSD est disponible via les portages *kfreebsd-amd64* et *kfreebsd-i386*, pour les PC 64 bits et 32 bits respectivement. Ces portages ont été publiés pour la première fois dans Debian 6.0 Squeeze comme *démonstration technique*.

De plus, le travail est en cours pour les portages suivants :

— *avr32*, portage pour l'architecture RISC 32 bits de Atmel,

— *hurd-i386*, un portage pour PC 32 bits. Ce portage utilise GNU Hurd, le système d'exploitation du projet GNU.

— *sh*, le portage vers les processeurs Hitachi SuperH.

Le projet de porter la distribution pour le noyau NetBSD, pour fournir *netbsd-i386* et *netbsd-alpha*, a été abandonné et aucun portage n'a jamais été publié.

Pour plus d'informations sur les portages disponibles, consultez les pages web sur les portages (`http://www.debian.org/ports/`).

4.3 Debian est-elle compatible avec les autres distributions ?

Les développeurs Debian communiquent avec les responsables des autres distributions Linux pour maintenir la compatibilité binaire entre les distributions Linux. La plupart des produits Linux commerciaux fonctionnent aussi bien sous Debian que sur le système pour lequel ils ont été construits.

Debian GNU/Linux se conforme à la norme de hiérarchie du système de fichiers Linux (`http://www.pathname.com/fhs/`) (« Linux Filesystem Hierarchy Standard »). Cependant, certaines des règles de cette norme laissent une part à l'interprétation, de ce fait il peut y avoir de légères différences entre un système Debian et les autres systèmes Linux.

Debian GNU/Linux prend en charge les logiciels développés pour Linux Standard Base (`http://www.linuxbase.org/`). LSB est une spécification visant à permettre l'exécution d'un unique paquet binaire par différentes distributions. La distribution Debian Etch est certifiée pour la version 3.1 de LSB, voir la page web des certifications de la Linux Foundation (`https://www.linux-foundation.org/lsb-cert/productdir.php`). Les discussions et la coordination visant la conformité de Debian aux exigences de la LSB se tiennent sur la liste de diffusion debian-lsb (`http://lists.debian.org/debian-lsb/`).

4.4 Comment le code source compatible Debian l'est-il avec d'autres systèmes Unix ?

Pour la plupart des applications, le code source de Linux est compatible avec les autres systèmes Unix. Il gère presque tout ce qui est disponible pour les systèmes Unix de type système V et pour les systèmes dérivés des BSD libres ou commerciaux. Cependant dans le monde Unix une telle déclaration n'a presque aucune valeur parce qu'il n'y a aucune possibilité de la prouver. Dans le secteur du développement logiciel, la compatibilité complète est exigée et pas seulement une compatibilité dans « la plupart des » cas. Ainsi il y a des années le besoin de normes s'est ressenti, et de nos jours POSIX.1 (norme 1003.1-1990 d'IEEE) est une des normes principales pour la compatibilité du code source des systèmes d'exploitation de type Unix.

Linux est prévu pour adhérer à POSIX.1, mais les normes de POSIX ne sont pas gratuites et la certification POSIX.1 (et FIPS 151-2) est très chère ; ce qui fait qu'il est très difficile pour les développeurs Linux de travailler sur une conformité complète de la norme Posix. Les coûts de certification rendent peu probable l'obtention par Debian d'une certification officielle de conformité même s'il passait complètement la suite de validation (la suite de validation est maintenant disponible gratuitement, ainsi on s'attend à ce que plus de personnes travaillent sur les questions POSIX.1).

Unifix GmbH (Braunschweig, Allemagne) a développé un système Linux qui a été certifié conforme à la norme FIPS 151-2 (un sur-ensemble de POSIX.1). Cette technologie était disponible dans une distribution de Unifix appelée Unifix Linux 2.0 et dans Linux-FT de Lasermoon.

4.5 Peut-on utiliser des paquets Debian (fichiers « .deb ») sur un système Linux Red Hat/Slackware/. . . ? Peux-t-on utiliser des paquets Red Hat (fichiers « .rpm ») sur un système Debian GNU/Linux ?

Les différentes distributions Linux utilisent des formats de paquets et des programmes de gestion de paquets différents.

Vous pouvez probablement : Un programme pour installer un paquet de Debian sur une machine Linux d'une autre distribution est disponible et fonctionnera généralement, dans le sens que les fichiers seront désarchivés. L'inverse est probablement vraie également, c.-à-d. : un programme pour désarchiver un paquet Red Hat ou Slackware sur une machine Debian GNU/Linux réussira probablement avec succès à désarchiver le paquet et à placer la plupart des fichiers dans leur répertoire prévu. C'est en grande partie une conséquence de l'existence (et de la large adhérence) à la norme de hiérarchie du système de fichiers Linux. Le paquet alien (`http://packages.debian.org/alien`) est utilisé pour convertir les différents formats des paquets.

Vous ne voulez probablement pas : La plupart des responsables de paquet écrivent des fichiers de contrôle utilisés pour le désarchivage des paquets. Ces fichiers de contrôle ne sont généralement pas standardisés. Par conséquent, l'effet de désarchiver un paquet Debian sur une machine d'une autre distribution aura (certainement) des effets imprévisibles sur le gestionnaire de paquets du système. De même, les utilitaires d'autres distributions pourraient réussir à dépaqueter leurs archives sur des systèmes Debian, mais feront probablement échouer le système de gestion de paquets de Debian quand le temps viendra de mettre à jour ou d'enlever quelques paquets, ou même simplement pour signaler exactement quels paquets sont présents sur un système.

Une meilleure méthode : La norme de système de fichiers Linux (et donc aussi de Debian GNU/Linux) exige que le sous-répertoire `/usr/local/` soit entièrement à la discrétion de l'utilisateur. Donc les utilisateurs peuvent désarchiver leur paquet dans ce répertoire, puis administrer leur configuration, les mettre à niveau et les déplacer individuellement.

4.6 Comment installer un programme n'appartenant pas à Debian ?

Les fichiers sous le répertoire `/usr/local/` ne sont pas sous le contrôle du système de gestion des paquets Debian. Ainsi, une bonne habitude est de mettre le code source de vos programmes dans le répertoire `/usr/local/src`. Par exemple, vous pouvez extraire les fichiers d'une archive appelée « foo.tar » dans le répertoire `/usr/local/src/foo`. Après la compilation, déposez les binaires dans `/usr/local/bin/`, les bibliothèques dans `/usr/local/lib/` et les fichiers de configuration dans `/usr/local/etc/`.

Si vos programmes et/ou fichiers doivent vraiment être placés dans un autre répertoire, vous pourriez les stocker tout de même dans `/usr/local/`, et établir les liens symboliques appropriés de l'endroit exigé vers son emplacement dans `/usr/local/`, par exemple, vous pourriez faire le lien

```
ln -s /usr/local/bin/foo /usr/bin/foo
```

De toute façon, si vous récupérez un programme dont le copyright permet la redistribution, vous devriez réfléchir à la possibilité d'en faire un paquet Debian et de l'envoyer au système de Debian. Les directives pour devenir un responsable de paquet sont incluses dans le manuel de la charte Debian (voir 'Quelles autres documentations existent sur le système Debian ?' page 53).

4.7 Pourquoi ne peut-on pas compiler de programme nécessitant libtermcap ?

Debian emploie la base de données de terminfo et la bibliothèque de ncurses pour les interfaces des terminaux, plutôt que la base de données et la bibliothèque de termcap. Les utilisateurs qui compilent des programmes demandant des connaissances sur l'interface terminal devraient remplacer les références à `libtermcap` par des références à `libncurses`.

Pour faire fonctionner les binaires qui ont déjà été liés avec la bibliothèque `termcap` et dont vous n'avez pas les sources, Debian fournit un paquet appelé `termcap-compat`. Ceci fournit les deux bibliothèques `libtermcap.so.2` et `/etc/termcap`. Installez ce paquet si le programme refuse de s'exécuter et affiche le message d'erreur « can't load library 'libtermcap.so.2' », ou se plaint au sujet du fichier `/etc/termcap` manquant.

4.8 Pourquoi ne peut-on pas installer AccelX ?

AccelX utilise la bibliothèque `termcap` pour son installation. Voir 'Pourquoi ne peut-on pas compiler de programme nécessitant libtermcap ?' de la présente page ci-dessus.

Chapitre 5

Logiciels disponibles sur un système Debian

5.1 Quelles sont les applications et les logiciels de développement disponibles sous Debian GNU/Linux ?

Comme pour la plupart des distributions Linux, Debian GNU/Linux fournit :

- la majorité des applications GNU pour le développement de logiciels, de manipulation de fichiers et de traitement de textes, y compris GCC, g++, make, texinfo, Emacs, le shell Bash et de nombreux utilitaires améliorés d'Unix,
- Perl, Python, Tcl/Tk, différents programmes apparentés, des modules et des bibliothèques pour chacun d'eux,
- TeX (LaTeX) et Lyx, dvips, Ghostscript,
- le système de fenêtrage Xorg, qui fournit pour Linux une interface utilisateur graphique gérée en réseau, et de nombreuses applications X comme les environnements de bureau GNOME, KDE et Xfce.
- un ensemble complet d'applications réseau, incluant des serveurs pour les protocoles internet (IP), telles que HTTP (WWW), FTP, NNTP (groupes de discussion), SMTP et POP (courrier) et DNS (serveur de noms) ; des bases de données relationnelles comme PostgreSQL, MyQSL ; des navigateurs web incluant les divers produits Mozilla [1],
- un ensemble complet d'applications de bureau, incluant la suite OpenOffice.org, Gnumeric et d'autres tableurs, des éditeurs WYSIWYG (« What You See Is What You Get », i.e. « vous obtenez ce que vous voyez »), des calendriers.

Plus de 28200 paquets, allant des serveurs, aux lecteurs de nouvelles, aux gestionnaires de son, aux programmes de fax, aux bases de données et aux tableurs, aux programmes de traitement d'image, de communication, de réseau, aux utilitaires de courrier, aux serveurs web, et même des programmes de radio amateur sont inclus dans la distribution. 187 autres logiciels sont disponibles sous forme de paquets Debian, mais ne font pas formellement partie de Debian à cause de leurs licences restrictives.

5.2 Qui a écrit tous ces logiciels ?

Pour chaque paquet les *auteurs* des programmes sont indiqués dans le fichier `/usr/share/doc/PAQUET/copyright`, où PAQUET doit être substitué par le nom du paquet.

Les *responsables* qui ont créé le paquet de ce logiciel pour le système Debian GNU/Linux sont listés dans le fichier de contrôle Debian (voir 'Qu'est ce qu'un fichier de contrôle ?' page 28) présent dans tous les paquets. Le changelog Debian, disponible à `/usr/share/doc/PACKAGE/changelog.Debian.gz`, mentionne aussi les personnes qui ont travaillé au paquetage Debian.

5.3 Comment puis-je obtenir la liste de tous les programmes qui ont été empaquetés pour Debian ?

Une liste complète est disponible sur n'importe quel miroir Debian (`http://www.debian.org/distrib/ftplist`), dans le fichier `indices/Maintainers`. Ce fichier contient le nom des paquets et les noms et adresses électroniques de leurs responsables respectifs.

1. Ils ont été, cependant, démarqués et sont fournis avec des noms différents à cause de problèmes liés à la marque commerciale

L'interface web pour les paquets Debian (http://packages.debian.org/) regroupe de façon pratique les paquets de l'archive Debian dans une trentaine de sections.

5.4 Comment puis-je installer un environnement de développeur pour construire des paquets ?

Si vous voulez construire des paquets pour votre système Debian, vous aurez besoin d'un environnement de développement basique, comprenant un compilateur C/C++ et quelques autres paquets essentiels. Afin d'installer cet environnement, vous avez juste besoin d'installer le `build-essential`. Ce paquet est un méta-paquet qui dépend des outils standards de développement dont on a besoin pour construire un paquet Debian.

Quelques logiciels peuvent cependant avoir besoin de programmes supplémentaires pour être reconstruits, comprenant les en-têtes de bibliothèque ou des outils additionnels tels que `autoconf` ou `gettext`. Debian fournit beaucoup des outils nécessaires pour construire d'autres logiciels en paquets Debian.

Trouver quel logiciel est précisément requis peut être embêtant. Cependant, à moins que vous n'ayez l'intention de reconstruire des paquets Debian. Cette dernière tâche est assez facile à faire, puisque les paquets officiels doivent inclure une liste des logiciels additionnels (en plus des paquets dans `build-essential`) nécessaires pour construire le paquet, connus sous le nom de `Build-Dependencies` (dépendances pour construire). Pour installer tous les paquets nécessaires à la construction d'un paquet source donné et ensuite construire le-dit paquet source vous pouvez juste lancer :

```
# apt-get build-dep foo
# apt-get source --build foo
```

Veuillez noter que si vous voulez construire les noyaux Linux distribués par Debian, vous voudrez également installer le paquet `kernel-package`. Pour plus d'informations, consultez 'Quel est l'outil fourni par Debian pour construire des noyaux personnalisés ?' page 47.

5.5 Que manque-t-il dans Debian GNU/Linux ?

Il existe une liste de paquets ayant encore besoin d'être empaquetés, les paquets en souffrance et paquets souhaités (http://www.debian.org/devel/wnpp/)

Pour plus d'informations sur l'ajout des choses manquantes, allez voir 'Comment devenir un développeur Debian ?' page 57.

5.6 Pourquoi est-ce que j'obtiens le message « ld : cannot find -lfoo » quand je compile un programme ? Pourquoi n'y a-t-il pas de fichier libfoo.so dans les paquets de bibliothèque Debian ?

La charte de Debian exige que de tels liens symboliques (sur libfoo.so.x.y.z ou semblable) soient placés dans des paquets séparés, les paquets de développement. Ces paquets sont habituellement appelés libfoo-dev ou libfooX-dev (si la bibliothèque est appelée libfooX où X est un nombre).

5.7 Comment Debian gère-t-elle Java ?

Quelques implémentations *libres* de la technologie Java sont disponibles comme paquets Debian, fournissant à la fois des kits de développement Java (JDK) et des environnments d'exécution (Runtime Environment). Vous pouvez écrire, déboguer et lancer des programmes Java en utilisant Debian.

L'exécution d'applets Java nécessite un navigateur web possédant les capacités de les identifier et de les exécuter. Plusieurs navigateurs web disponibles dans Debian, tels que Mozilla ou Konqueror gèrent les greffons Java qui permettent l'exécution des applets Java.

Reportez-vous à la FAQ Java et Debian GNU/Linux (http://www.debian.org/doc/manuals/debian-java-faq/) pour plus d'informations.

5.8 Comment puis-je vérifier que j'emploie un système Debian, et quelle est sa version ?

Afin de s'assurer que votre système a été installé à partir des disques d'installation de Debian, vérifiez l'existence du fichier `/etc/debian_version`, qui contient une seule ligne donnant le numéro de version, défini par le paquet `base-files`.

L'existence sur votre système du programme `dpkg` prouve que vous devriez pouvoir installer des paquets Debian, mais comme le programme a été porté sur beaucoup d'autres systèmes d'exploitation et d'architectures, ce n'est donc pas une méthode fiable pour déterminer si c'est un système Debian GNU/Linux.

Les utilisateurs devraient se rendre compte, cependant, que le système de Debian est constitué de beaucoup de composants, dont chacun peut être mis à jour (presque) indépendamment. Chaque version de Debian possède un contenu bien défini et invariable. Les mises à jour sont disponibles séparément. Pour avoir sur une ligne la description de l'état d'installation du paquet `foo`, utilisez la commande `dpkg --list foo`. Pour voir les numéros de versions de tous les paquets installés, exécutez :

```
dpkg -l
```

Pour une description plus détaillée, employez :

```
dpkg --status foo
```

5.9 Comment Debian gère-t-elle les langues non anglophones ?

— Debian GNU/Linux est distribuée avec des cartes de clavier (keymaps) pour presque deux douzaines de clavier, et avec des utilitaires (dans le paquet `kbd`) pour installer, regarder, et modifier ces tables.
  Lors de l'installation, l'utilisateur doit indiquer le clavier qu'il emploiera.

— La grande majorité des logiciels que nous avons empaquetés gèrent la saisie de caractères non-US-ASCII utilisés dans d'autres langues latines (par exemple iso-8859-1 ou iso-8859-2), et un certain nombre de programmes gèrent les langues sur plusieurs octets telles que le japonais ou le chinois.

— Actuellement, la gestion pour les pages de manuel est fournie en allemand, espagnol, finnois, français, hongrois, italien, japonais, coréen, hollandais, polonais, portugais, russe, turc et chinois grâce au paquet `manpages-LANG` (où LANG est constitué des deux lettres du code ISO du pays). Pour accéder à une page de manuel NLS, l'utilisateur doit positionner la variable d'environnement LC_MESSAGES correctement.
  Par exemple, dans le cas des pages de manuel en français, LC_MESSAGES doit être positionnée à « french ». Le programme `man` recherchera alors les pages de manuel en français sous /usr/share/man/fr/.

5.10 Où est le programme pine ?

En raison de sa licence restrictive, il est dans la section « non-free ». D'ailleurs, puisque la licence ne permet même pas de distribuer des binaires modifiés, vous devrez le compiler vous-même avec les sources et les correctifs Debian.

Le paquet source s'appelle `pine`. Vous pouvez utiliser le paquet `pine-tracker` pour être informé quand vous devez le mettre à jour.

Notez qu'il existe beaucoup de remplaçants pour les programmes pine et pico, tels que `mutt` et `nano`, qui eux sont situés dans la section « main ».

5.11 Où sont les programmes qmail, ezmlm et djbdns ?

Dan J. Bernstein publiait tous les logiciels qu'il développe (http://cr.yp.to/software.html) avec une licence restrictive qui n'autorisait pas à modifier les binaires distribués. Cependant, en novembre 2007, Bernstein a déclaré : « [...] j'ai décidé de mettre tous mes programmes futurs et [...] passés dans le domaine public. » Consultez la FAQ des distributeurs (http://cr.yp.to/distributors.html) pour les termes de sa distribution.

Depuis septembre 2008, `daemontools`, `djbdns` et `ucspi-tcp` sont distribués dans Debian Lenny (dans la section main). Au moment où ces lignes ont été écrites (septembre 2008), ni `qmail` ni `ezmlm` ne sont encore empaquetés pour Debian main ; consultez Bug #457318 (ITP qmail) (http://bugs.debian.org/457318) et Bug #469193 (ITP ezmlm-idx) (http://bugs.debian.org/469193) pour connaître l'état actuel de ces paquets.

En septembre 2008, `publicfile` n'est pas encore un logiciel libre.

5.12 Où y a-t-il un lecteur pour Flash (SWF) ?

Debian contient `gnash` et `swfdec` : 2 lecteurs libres de vidéos SWF.

5.13 Où est Google Earth ?

Google Earth, disponible pour GNU/Linux à partir du site web de Google, n'est non seulement pas un logiciel libre, mais est complètement non distribuable par un tiers. Cependant, `googleearth-package` (dans la section contrib) peut aider à utiliser ce logiciel.

5.14 Où est le logiciel de VoIP ?

Deux protocoles ouverts principaux sont utilisés pour la voix sur IP : SIP et H.323. Les deux sont implémentés par une large variété de programmes dans la section main de Debian. `ekiga` est l'un des clients les plus populaires.

5.15 J'ai une carte réseau sans-fil qui ne fonctionne pas avec Linux. Que dois-je faire ?

Achetez-en une qui fonctionne. :-)

Autrement, utilisez `ndiswrapper` pour utiliser un pilote pour Windows (si vous en avez un) avec votre système Linux. Consultez la page ndiswrapper du Wiki Debian (http://wiki.debian.org/NdisWrapper) pour plus d'informations.

Chapitre 6

Les archives FTP de Debian

6.1 Combien de distributions de Debian y a-t-il ?

Il y a trois distributions majeures : la distribution « stable », la distribution « testing », et la distribution « unstable ». La distribution « testing » est quelque fois gelée (« frozen ») (voyez ‘Que dire de « testing » ? Comment est-elle gelée (« frozen ») ?’ page 23). À côté de celles-là, on trouve également la distribution « oldstable » (celle qui précédait la « stable ») ainsi que la distribution « experimental ».

« Experimental » est utilisée pour les paquets encore en cours de développement qui comportent des risques importants pour le fonctionnement de votre système. Elle est utilisée par les développeurs qui veulent étudier et tester les versions les plus récentes des logiciels. Les utilisateurs ne devraient pas utiliser ces paquets-là, car ils peuvent être dangereux et endommager les systèmes des utilisateurs même les plus expérimentés.

Consultez ‘Choisir une distribution de Debian’ page 7 pour de l’aide sur le choix d’une distribution Debian.

6.2 À quoi correspondent tous les noms comme etch, lenny, etc. ?

Ce sont justes des noms de code. Quand une distribution Debian est en cours de développement, elle n’a aucun numéro de version mais un nom de code. Le but de ces noms de code est de faciliter la copie sur les miroirs des distributions Debian (si un véritable répertoire comme `unstable` est soudainement renommé en `stable`, beaucoup de choses devraient être inutilement téléchargées).

Actuellement, la version `stable` est un lien symbolique vers `wheezy` (c.-à-d. Debian GNU/Linux 7.0) et la version `testing` est un lien symbolique vers `jessie`. Ceci signifie que `wheezy` est la distribution « stable » actuelle et `jessie` est la distribution « testing » actuelle.

`Unstable` est un lien symbolique permanent vers `sid`, car `sid` est toujours la distribution « unstable ».

6.2.1 Quels noms de code ont déjà été utilisés ?

Les autres noms de code qui ont déjà été employés sont : `buzz` pour la version 1.1, `rex` pour la version 1.2, `bo` pour les versions 1.3.x, `hamm` pour la version 2.0, `slink` pour la version 2.1, `potato` pour la version 2.2, `woody` pour la version 3.0, `sarge` pour la version 3.1, `etch` pour la version 4.0, `lenny` pour la version 5.0, `squeeze` pour la version 6.0 et `wheezy` pour la version 7.0.

6.2.2 D’où proviennent les noms de code ?

Jusqu’ici les noms de code proviennent des personnages des films « Toy Story » par Pixar.

— *buzz* (Buzz Lightyear) est le cosmonaute,
— *rex* est le tyrannosaure,
— *bo* (Bo Peep) est la bergère,
— *hamm* est la tirelire en forme de cochon,
— *slink* (Slinky Dog) est le chien,

— *potato* est bien sûr, Mr. Patate,
— *woody* est le cowboy,
— *sarge* est le sergent de l'armée de plastique vert,
— *etch* est l'ardoise magique (Etch-a-Sketch),
— *lenny* est la paire de jumelles,
— *squeeze* est le nom des extraterrestres à 3 yeux,
— *wheezy* est le nom du manchot au n½ud papillon rouge,
— *jessie* est l'écuyère.
— *sid* est le garçon des voisins qui détruit les jouets.

6.3 Que dire de « sid » ?

Sid ou « *unstable* » est le lieu où la plupart des paquets sont initialement envoyés. Elle ne sera jamais directement publiée, parce que les paquets devront d'abord être inclus dans « *testing* », afin d'être publiés dans « *stable* » plus tard. Sid contient des paquets pour l'ensemble des architectures publiées ou non.

Le nom « sid » vient aussi du film d'animation Toy Story : Sid est le garçon des voisins qui détruit les jouets.

[1]

6.4 Que contient le répertoire stable ?

— stable/main/ : Ce répertoire contient les paquets qui constituent la version la plus récente du système Debian GNU/Linux.
  Ces paquets sont tous conformes aux principes du logiciel libre selon Debian (http://www.debian.org/social_contract#guidelines) (Debian Free Software Guidelines, DFSG) et sont tous librements utilisables et librements distribuables.
— stable/non-free/ : Ce répertoire contient les paquets de la distribution ayant certaines restrictions, ce qui oblige les distributeurs à tenir compte soigneusement des conditions définies dans les copyright.
  Par exemple, certains paquets ont une licence avec une clause interdisant une distribution commerciale. D'autres peuvent être redistribués mais sont en fait des partagiciels (shareware) et non pas des logiciels libres (free software). La licence de chaque paquet doit être étudiée, et probablement négociée, avant qu'ils ne soient inclus dans toutes distributions (par exemple sur un CD-ROM).
— stable/contrib/ : Ce répertoire contient les paquets qui sont conformes aux DFSG et *librement distribuables*, mais dépendent d'une façon ou d'une autre d'un paquet qui *n'est pas* librement distributable et ainsi disponible seulement dans la section non-free.

6.5 Que contient la distribution testing ?

Des paquets sont installés dans le répertoire « testing » après qu'ils aient subi un certain nombre de tests dans « unstable ».

Ils doivent être synchronisés sur toutes les architectures où ils ont été construits et ne doivent pas avoir de dépendances qui empêcheraient leur installation ; ils doivent également avoir moins de bogues critiques que les versions actuellement dans « testing ». De cette façon, nous espérons que « testing » soit une version toujours prête à la publication.

Plus d'informations sur l'état de « testing » en général et sur les différents paquets sont disponibles à http://www.debian.org/devel/testing.

1. Quand la sid n'existait pas, l'organisation du site FTP avait un défaut majeur. On supposait que quand une architecture était créée dans la distribution « unstable » actuelle, elle serait publiée quand la distribution deviendrait la nouvelle distribution « stable ». Pour beaucoup d'architectures ce n'était pas le cas, ce qui obligeait à déplacer ces répertoires lors de la sortie d'une version. Ce n'était pas pratique parce que le déplacement aurait consommé beaucoup de bande passante. Les administrateurs ont évité le problème pendant plusieurs années en plaçant les binaires pour les architectures non publiées dans un répertoire particulier nommé « sid ». Pour les architectures non encore publiées, lors de leurs sorties, un lien entre la stable courante et sid était créé et à partir de là elles étaient créées normalement dans l'arborescence « unstable ». Cette disposition était légèrement déconcertante pour les utilisateurs. Avec l'arrivée des répertoires communs (voir 'Que trouve-t-on dans le répertoire pool ?' page 24), les paquets ont commencé à être placés dans un endroit standard dans le pool, quelle que soit la distribution, donc la publication d'une distribution n'entraîne plus de consommation excessive de bande passante sur les miroirs (mais il y a cependant une consommation graduelle de la bande passante pendant le processus de développement).

6.5.1 Que dire de « testing » ? Comment est-elle gelée (« frozen ») ?

Quand la distribution « testing » est suffisamment mature, le reponsable de la publication commence à geler la distribution. Le temps de propagation des paquets entre les distributions est augmenté pour s'assurer que le moins possible de bogues passe de la distribution « unstable » dans « testing ».

Après un moment, la distribution « testing » devient vraiment gelée. Ceci signifie que tous les nouveaux paquets qui devaient entrer dans « testing » sont bloqués, à moins qu'ils ne corrigent un bogue critique (release critical). La distribution « testing » peut également demeurer dans un gel profond pendant les cycles d'essai, quand la publication est imminente.

Lorsque la distribution « testing » est « gelée », « unstable » a tendance à geler également. En effet, les développeurs sont réticents à envoyer des logiciels complètement nouveaux dans l'archive : cela compliquerait le processus de correction dans le cas où un logiciel dans « testing » doit être corrigé, suite à un bogue mineur voire critique pour la publication.

Nous conservons un enregistrement des bogues de la distribution « testing » qui peuvent empêcher un paquet d'être publié, ou retarder la publication de la distribution. Pour plus de détails, veuillez-vous reporter aux informations sur la version testing actuelle (http://www.debian.org/releases/testing/).

Une fois que le nombre de bogues a atteint une valeur maximale acceptable, la distribution « testing » gelée est déclarée « stable » et publiée avec un numéro de version.

Le plus important des décomptes de bogues est celui des bogues critiques pour la publication (« Release Critical ») (http://bugs.debian.org/release-critical/). Un objectif courant pour la publication est de réduire à zéro le nombre de bogues critiques, graves ou sérieux (NoRCBugs (http://wiki.debian.org/ReleaseGoals/NoRCBugs)). La liste complète des problèmes considérés critiques est disponible dans le document de politique des bogues RC (http://release.debian.org/testing/rc_policy.txt).

Avec chaque nouvelle version, l'ancienne distribution « stable » devient obsolète et est déplacée de l'archive. pour plus d'informations, reportez-vous à Debian archive (http://www.debian.org/distrib/archive).

6.6 Que contient la distribution unstable ?

Le répertoire « unstable » contient une image du système en cours de développement. Les utilisateurs sont les bienvenus pour utiliser et tester ces paquets, mais soyez averti au sujet de leur état. L'avantage d'utiliser la distribution « unstable » est que votre système est toujours à jour avec la dernière version des logiciels GNU/Linux, mais s'il y a un problème, vous en découvrirez les mauvais côtés.

Il y a aussi dans « unstable » des sous-répertoires main, contrib et non-free remplis avec les mêmes critères que dans « stable ».

6.7 Quels sont tous ces répertoires dans les archives FTP de Debian ?

Le logiciel empaqueté pour Debian GNU/Linux est disponible dans un des nombreux répertoires présent sur chaque site miroir de Debian.

Le répertoire `dists` est une abréviation pour « distributions » et c'est la manière canonique pour accéder à la version (et pré-versions) Debian actuellement disponible.

Le répertoire `pool` contient les paquets réels, voir 'Que trouve-t-on dans le répertoire `pool` ?' page suivante.

On trouve aussi les répertoires supplémentaires suivant :

/tools/ **:** Utilitaire DOS pour la création de disque de démarrage, pour le partitionnement de votre disque, pour la compression et la décompression de fichiers et pour le démarrage de Linux.

/doc/ **:** La documentation de base de Debian, comme cette FAQ, les instructions pour le système de rapport de bogues, etc.

/indices/ **:** Différents fichiers d'index (comme le fichier Maintainers et les fichiers override).

/project/ **:** Principalement des ressources pour les développeurs et divers fichiers.

6.8 Que sont tous ces répertoires dans `dists/stable/main` ?

Dans chacune des arborescences des répertoires majeurs[2], il y a trois ensembles de sous-répertoires contenant les fichiers catalogues.

2. `dists/stable/main`, `dists/stable/contrib`, `dists/stable/non-free` et `dists/unstable/main/`, etc.

Il y a un ensemble de sous-répertoires `binary-quelquechose` contenant les fichiers catalogues pour les paquets binaires de chaque architecture disponible. Par exemple, `binary-i386` pour les paquets s'exécutant sur les machines Intel x86 ou `binary-sparc` pour les paquets s'exécutant sur les SPARCStations Sun.

La liste complète de toutes les architectures disponibles pour chaque version est accessible à l'adresse the release's web page (http://www.debian.org/releases/). Pour la version en cours, veuillez-vous reporter à 'Sur quelle architecture matérielle fonctionne Debian GNU/Linux ?' page 13.

Les fichiers catalogues dans les répertoires `binary-*` sont nommés Packages(.gz, .bz2) et fournissent un résumé de chaque paquet binaire présent dans la distribution. Les paquets binaires se trouvent à la racine du répertoire `pool`.

De plus, il y existe un sous-répertoire nommé `source/` contenant les fichiers catalogues pour les paquets sources contenus dans la distribution. Le fichier catalogue est nommé Sources(.gz, .bz2).

Le dernier mais non des moindre, est un ensemble de sous-répertoires utilisé pour les fichiers catalogues du système d'installation. Ils sont présents dans un répertoire `debian-installer/binary-architecture`.

6.9 Où trouve-t-on le code source ?

Le code source est inclus pour tout le système Debian. De plus, les termes de la licence de la plupart des programmes du système *requièrent* que le code source soit distribué avec le programme, ou qu'un moyen de récupérer ce code accompagne le programme.

Le code source est distribué dans le répertoire `pool` (voir 'Que trouve-t-on dans le répertoire `pool` ?' de la présente page) avec les binaires de toutes les architectures. Pour récupérer le code source sans être familier avec la structure de l'archive FTP, exécutez une commande comme `apt-get source monpaquet`.

Certains paquets ne sont distribués que sous forme de code source à cause de restrictions de leur licence. Notamment, l'un de ces paquets est `pine`, voir 'Où est le programme pine ?' page 19 pour plus d'informations.

Le code source peut ou non être disponible pour les paquets dans les sections « contrib » et « non-free », qui ne font pas formellement partie du système Debian.

6.10 Que trouve-t-on dans le répertoire `pool` ?

Les paquets sont gardés dans un répertoire commun (« pool »), structuré selon le nom des paquets sources. Pour rendre cela gérable, le répertoire est divisé par section (« main », « contrib » et « non-free ») et dans chaque section par la première lettre du nom des paquets sources. Ces répertoires contiennent plusieurs fichiers : les paquets binaires pour chaque architecture et les paquets sources à partir desquels sont construits les paquets binaires.

Vous pouvez voir où chaque paquet est conservé en exécutant la commande `apt-cache showsrc monpaquet` et en regardant la ligne « Directory : ». Par exemple, les paquets `apache` sont conservés dans `pool/main/a/apache/`.

De plus, comme il existe de nombreux paquets de bibliothèque `lib*`, ceux-ci sont traités différemment. Par exemple, le paquet `libpaper` est placé dans le répertoire `pool/main/libp/libpaper/`.

[3]

6.11 Qu'est-ce que le répertoire incoming ?

Après qu'un développeur a envoyé un paquet, il est conservé dans le répertoire « incoming » avant de vérifier son origine et de l'autoriser dans l'archive.

Généralement personne ne devrait installer des paquets provenant de ce répertoire. Cependant, dans certains rares cas d'urgence le répertoire « incoming » est disponible à http://incoming.debian.org/. Vous devrez télécharger manuellement les paquets, vérifier les signatures GPG et les sommes MD5 dans les fichiers .changes et .dsc et les installer ensuite.

3. Historiquement, les paquets étaient conservés dans le sous-répertoire `dists` correspondant aux distributions. Ceci s'est avéré poser certains problèmes, comme une grande consommation de bande passante sur des miroirs lors de changements majeurs. Ces problèmes ont été résolus avec l'introduction du répertoire pool. Le répertoire `dists` est encore utilisé pour les fichiers catalogues servant à des programmes comme `apt`.

6.12 Comment mettre en place son propre dépôt pour apt ?

Si vous avez construit quelques paquets Debian que vous voudriez pouvoir installer en utilisant les outils de gestion de paquets de Debian, vous pouvez mettre en place votre propre archive de paquets pour apt. C'est également utile si vous voulez partager vos paquets Debian tant qu'ils ne sont pas distribués par le projet Debian. Les instructions de mise en oeuvre sont données dans le Guide pratique des référentiels Debian (http://www.debian.org/doc/manuals/repository-howto/repository-howto).

Chapitre 7

Les bases du système de gestion des paquets Debian

Ce chapitre touche aux mécanismes internes de bas niveau du système de gestion de paquets Debian. Si vous êtes avant tout intéressé par l'*utilisation* des outils appropriés, passez aux chapitres 'Outils de gestion des paquets Debian' page 35 et 'Garder son système Debian à jour' page 43.

7.1 Qu'est ce qu'un paquet Debian ?

Les paquets contiennent généralement tous les fichiers nécessaires pour implémenter un ensemble de commandes ou de fonctionnalités. Il y a deux sortes de paquets Debian :

— *Les paquets binaires* contenant les exécutables, les fichiers de configuration, les pages de manuel ou d'info, les informations de copyright et d'autres documentations. Ces paquets sont distribués sous un format d'archive spécifique à Debian (voir 'Quel est le format d'un paquet binaire Debian ?' de la présente page). Ils sont habituellement reconnaissables par l'extension « .deb ». Ils peuvent être installés en utilisant l'utilitaire `dpkg` (éventuellement avec une interface comme `aptitude`) ; vous trouverez plus de détails dans les pages de manuel.

— *Les paquets sources* sont constitués d'un fichier `.dsc` décrivant le paquet source (incluant le nom des fichiers suivants), un fichier `.orig.tar.gz` contenant les sources originales non modifiés au format tar compressé et habituellement un fichier `.diff.gz` contenant les modifications spécifiques à Debian par rapport au source original. L'utilitaire `dpkg-source` permet l'archivage et le désarchivage des sources Debian ; vous trouverez plus de détails dans les pages de manuel. (Le programme `apt-get` peut être utilisé comme une interface pour `dpkg-source`.)

L'installation de logiciels par le système de paquets utilise les dépendances qui sont définies avec attention par le responsable du paquet. Ces dépendances sont documentées dans le fichier `control` associé à chaque paquet. Par exemple, le paquet du compilateur GNU C (`gcc`) dépend du paquet `binutils` qui contient l'éditeur de lien et l'assembleur. Si un utilisateur essaye d'installer `gcc` sans avoir préalablement installé `binutils`, le système de gestion de paquets (dpkg) renverra un message d'erreur disant qu'il a aussi besoin de `binutils` et arrêtera l'installation de `gcc`. (Cependant ceci peut être forcé par un utilisateur insistant, voir `dpkg(8)`). Pour plus d'informations, voir 'Que signifie qu'un paquet dépend (*Depends*), recommande (*Recommends*), suggère (*Suggests*), est en conflit (*Conflicts*), remplace (*Replaces*), casse (*Breaks*) ou fournit (*Provides*) un autre paquet ?' page 30 plus loin.

Les outils de gestion de paquets Debian peuvent être utilisés pour :

— manipuler ou administrer les paquets ou une partie des paquets,
— administrer les modifications locales (« overrides ») des fichiers d'un paquets,
— aider les développeurs dans la construction de paquets et
— aider les utilisateurs dans l'installation de paquets résidant sur un serveur FTP distant.

7.2 Quel est le format d'un paquet binaire Debian ?

Un paquet Debian ou un fichier d'archive Debian, contient les fichiers exécutables, les bibliothèques, et la documentation liée à une suite particulière de programme ou à un ensemble de programmes. Normalement, un fichier archive Debian a un nom se terminant par `.deb`.

Les mécanismes internes du format des paquets binaires Debian sont décrits dans la page de manuel de `deb(5)`. Ce format interne est susceptible de changer (entre chaque version principale de Debian GNU/Linux), donc utilisez toujours `dpkg-deb(1)` si vous avez besoin d'effectuer des opérations de bas niveau sur des fichiers `.deb`.

7.3 Pourquoi les noms des paquets Debian sont-ils si longs ?

Le nom des paquets binaires Debian se conforme à la convention suivante : <foo>_<NuméroVersion>-<NuméroRévisionDebian>_<DebianArchitecture>.deb

Notez que `foo` est supposé être le nom du paquet. On peut connaître le nom associé à un fichier d'archive Debian particulier (fichier .deb) d'une de ces manières :

— inspecter le fichier « Packages » dans le répertoire où il a été stocké sur un site FTP d'archive Debian. Ce fichier contient une description pour chaque paquet ; le premier champ de chaque description est le nom formel du paquet.

— utiliser la commande `dpkg --info foo_VVV-RRR_AAA.deb` (où VVV, RRR et AAA sont respectivement le numéro de version, le numéro de révision, et l'architecture du paquet en question). Ceci affiche, entre autres, le nom du paquet correspondant au fichier archive.

Le composant `VVV` est le numéro de version fourni par le développeur amont. Il n'y a pas de standard ici, donc le numéro de version peut avoir différents formats comme « 19990513 » ou « 1.3.8pre1 ».

Le composant `RRR` est le numéro de révision Debian et il est choisi par le développeur Debian (ou un utilisateur s'il choisit de construire lui-même son paquet). Ce numéro correspond au niveau de révision du paquet Debian, ainsi, une nouvelle révision signifie généralement des modifications du Makefile Debian (`debian/rules`), du fichier de contrôle Debian `debian/control`), des scripts d'installation ou de désinstallation (`debian/p*`) ou des fichiers de configurations utilisés avec le paquet.

Le composant `AAA` identifie le processeur pour lequel le paquet a été construit. Il s'agit généralement de `i386`, pour les processeurs compatibles avec l'Intel 386 et ultérieurs. La structure de l'archive Debian 'Quels sont tous ces répertoires dans les archives FTP de Debian ?' page 23 montre les autres possibilités. Pour plus de détails, consultez la description de « l'architecture Debian » dans la page de manuel `dpkg-architecture(1)`.

7.4 Qu'est ce qu'un fichier de contrôle ?

Des détails concernant le contenu du fichier `control` Debian sont fournis dans la charte Debian, au chapitre cinq, voir 'Quelles autres documentations existent sur le système Debian ?' page 53.

Brièvement, un fichier `control` est montré ci-dessous pour le paquet Debian hello :

```
Package: hello
Priority: optional
Section: devel
Installed-Size: 45
Maintainer: Adam Heath <doogie@debian.org>
Architecture: i386
Version: 1.3-16
Depends: libc6 (>= 2.1)
Description: Le classique et bon exemple
 Le programme bonjour de GNU produit une salutation familière et amicale. Il

 permet à des non-programmeurs d'utiliser un outil classique d'informatique
 qui leur serait autrement indisponible.
 .
 Sérieusement:  c'est un exemple sur comment faire un paquet Debian.
 C'est la version Debian du programme du projet GNU «~bonjour
 le monde~» (qui est lui-même un exemple pour le projet de GNU).
```

Le champ « Package » contient le nom du paquet. C'est le nom par lequel les outils de gestion de paquet peuvent le manipuler. Il est habituellement similaire mais pas nécessairement le même à la première chaîne composant le nom de l'archive Debian.

Le champ de « Version » donne le numéro de version du développeur amont et (dans le dernier composant) le niveau de révision du paquet de Debian de ce programme comme expliqué dans 'Pourquoi les noms des paquets Debian sont-ils si longs ?' de la présente page.

Le champ « Architecture » indique pour quel processeur ce binaire particulier a été compilé.

Le champ « Depends » donne une liste des paquets qui doivent être installés afin d'installer ce paquet avec succès.

Le champ « Installed-Size » indique l'espace disque utilisé par le paquet une fois installé. Ceci est prévu pour être employé par les programmes d'installation pour vérifier s'il y a suffisamment d'espace disque disponible pour installer le programme.

La ligne « Section » indique la section où le paquet Debian est conservé sur les sites FTP de Debian.

Le champ « Priority » indique l'importance d'un paquet pour l'installation, de sorte que des logiciels semi-intelligents comme dselect ou aptitude peuvent trier les paquets dans une catégorie, par exemple les paquets installés facultativement. Voir 'Qu'est ce qu'un paquet *Essential*, *Required*, *Important*, *Standard*, *Optional* ou *Extra* ?' de la présente page.

Le champ « Maintainer » contient l'adresse électronique de la personne actuellement responsable de la maintenance du paquet.

Le champ « Description » fournit un bref résumé des fonctionnalités du paquet.

Pour plus d'informations sur les champs disponibles dans un paquet, reportez-vous au chapitre 5 de la charte Debian, « Fichiers control et leurs champs », consultez 'Quelles autres documentations existent sur le système Debian ?' page 53.

7.5 Qu'est ce qu'un Debian conffile ?

Conffiles est une liste de fichiers de configuration (habituellement placés dans `/etc`) que le système de gestion de paquet n'écrasera pas quand le paquet est mis à jour. Ceci s'assure que les valeurs locales pour ces fichiers seront préservées, et c'est un dispositif critique permettant la mise à niveau des paquets sur un système en fonctionnement.

Pour déterminer exactement quels fichiers sont préservés pendant la mise à jour, lancez :

```
dpkg --status package
```

et regardez la section « Conffiles : ».

7.6 Qu'est-ce qu'un script Debian preinst, postinst, prerm, et postrm ?

Ces fichiers sont des scripts exécutables qui sont lancés automatiquement avant ou après l'installation d'un paquet. Avec le fichier appelé `control`, tous ces fichiers font partie de la section commande d'une archive Debian.

Les fichiers individuels sont :

preinst Ce script est exécuté avant que le paquet soit désarchivé du fichier archive (.deb). Beaucoup de scripts « preinst » arrêtent les services des paquets qui sont mis à jour jusqu'à ce que leurs installations ou leurs mises à jour soient terminées (suivant le succès de l'exécution du script « postinst »).

postinst Ce script termine typiquement n'importe quelle configuration exigée du paquet `foo` une fois que `foo` a été désarchivé à partir du fichier d'archive Debian (« deb »). Souvent le script « postinst » demande à l'utilisateur des informations et avertit l'utilisateur que s'il accepte les valeurs par défaut, il devrait se souvenir de revenir en arrière et de reconfigurer le paquet lorsque la situation le demandera. Beaucoup de scripts « postinst » exécutent toutes les commandes nécessaires au démarrage ou au redémarrage d'un service une fois qu'un nouveau paquet a été installé ou mis à jour.

prerm Ce script arrête généralement tous les démons qui sont associés avec le paquet. Il est exécuté avant la suppression des fichiers associés au paquet.

postrm Ce script modifie généralement les liens ou les autres fichiers associés à `foo` et/ou supprime les fichiers créés par le paquet (Voir aussi 'Qu'est-ce qu'un paquet virtuel ?' page suivante.)

Actuellement tous les fichiers de commande peuvent être trouvés dans le répertoire `/var/lib/dpkg/info`. Les fichiers concernant le paquet `foo` commencent par le nom "foo" et sont suivis par les extensions "preinst", "postinst" etc. Le fichier `foo.list` dans ce répertoire liste tous les fichiers installés par le paquet `foo`. (Notez que l'emplacement de ces fichiers est interne à dpkg ; vous ne devriez pas compter sur eux).

7.7 Qu'est ce qu'un paquet *Essential*, *Required*, *Important*, *Standard*, *Optional* ou *Extra* ?

À chaque paquet Debian est assignée une priorité par les développeurs de la distribution, c'est une aide pour le système de gestion de paquet. Les priorités sont :

— Nécessaire (**Required**) : paquets nécessaire pour le fonctionnement correct du système.
  Ceci inclut tous les outils nécessaires pour la réparation d'un système défectueux. Vous ne devez pas supprimer ces paquets ou votre système risque de devenir complètement instable et vous ne pourrez probablement pas utiliser dpkg pour corriger cela. Avec seulement les paquets nécessaires, le système sera probablement inutilisable mais il aura suffisamment de fonctionnalités pour permettre de démarrer et d'installer plus de logiciels.
— **Important** : paquets devant être présents sur tous systèmes de type Unix
  Les autres paquets sans lesquels le système ne fonctionnera pas correctement ou ne sera pas utilisable ont cette priorité. Ceci n'inclut *PAS* Emacs, X, TeX ou aucune autre grosse application. Ces paquets constituent seulement l'infrastructure de base.
— **Standard** : paquets présents sur n'importe quel système Linux, y compris un système raisonnablement petit mais pas limité à un système en mode console. Des outils sont disponibles pour naviguer sur Internet (en utilisant w3m), envoyer des courriels (avec mutt) et télécharger des fichiers depuis des serveurs FTP.
  C'est ce qui sera installé par défaut si les utilisateurs ne sélectionnent rien. Cela n'inclut pas les grosses applications, mais intègre l'interpréteur Python et certains serveurs comme OpenSSH (pour l'administration à distance), Exim (pour la livraison des courriels, même s'il peut être configuré en mode local uniquement), un serveur identd (pidentd) et le portmapper RPC (`portmap`). Sont également intégrées certaines documentations générales considérées utiles par la plupart des utilisateurs.
— Optionnel (**Optional**) : paquets incluant tous ce que vous pourriez raisonnablement vouloir installer si vous ne savez pas ce que c'est ou si vous n'avez pas précisé de condition.
  Ceci inclut X, une distribution TeX complète et beaucoup d'applications.
— **Extra** : paquets qui entrent en conflit avec d'autres ayant des priorités plus élevées ou ayant des conditions particulières qui les rendent peu convenables pour être optionnels. Ils sont utiles seulement si vous savez déjà ce qu'ils font.

Si vous faites une installation Debian par défaut, tous les paquets ayant une priorité **standard** ou supérieure seront installés sur votre système. Si vous choisissez des tâches prédéfinies, vous obtiendrez aussi les paquets de priorité plus faible.

De plus, certain paquets sont marqués comme essentiels (**Essential**) puisqu'ils sont absolument nécessaires pour un fonctionnement correct du système. Les outils de gestion de paquets refuseront de les supprimer.

7.8 Qu'est-ce qu'un paquet virtuel ?

Un paquet virtuel est un nom générique qui s'applique à n'importe quel groupe de paquets fournissant une fonctionnalité de base semblable. Par exemple, les programmes `tin` et `trn` sont des programmes de lecture de nouvelles Usenet et devraient donc satisfaire n'importe quelle dépendance d'un programme exigeant la présence d'un programme de lecture de nouvelles sur le système, afin de fonctionner ou pour être utiles. Les deux sont donc capables de fournir le paquet virtuel appelé `news-reader`.

De la même façon, `smail` et `sendmail` fournissent tous les deux les fonctionnalités d'un agent de transport de courrier électronique. Les deux sont donc capables de fournir le paquet virtuel `mail-transport-agent`. Si l'un ou l'autre est installé, alors n'importe quel programme exigeant l'installation d'un agent de transport de courrier électronique sera satisfait par l'existence de ce paquet virtuel.

Debian fournit un mécanisme de sorte que, si plus d'un paquet fournissant le même paquet virtuel est installé sur le système, alors les administrateurs peuvent choisir leur paquet préféré. La commande appropriée est `update-alternatives`, et est décrite plus loin dans 'Certains utilisateurs apprécient mawk, d'autres gawk ; certains utilisent vim, d'autres elvis ; certains préfèrent trn, d'autres tin ; comment Debian gère-t-elle la diversité ?' page 52.

7.9 Que signifie qu'un paquet dépend (*Depends*), recommande (*Recommends*), suggère (*Suggests*), est en conflit (*Conflicts*), remplace (*Replaces*), casse (*Breaks*) ou fournit (*Provides*) un autre paquet ?

Le système de paquet Debian a une gamme de « dépendances » sur les paquets qui permet d'indiquer (avec un simple drapeau) à quel niveau un programme A peut fonctionner indépendamment de l'existence du programme B sur un système donné :

— Le paquet A *dépend* du paquet B si B doit absolument être installé pour pouvoir exécuter A. Dans certains cas, A dépend non seulement de B mais d'une version de B. Dans ce cas, la dépendance de version est une limite basse, dans le sens que A dépend de toutes versions plus récentes que celle spécifiée.

— Le paquet A *recommande* le paquet B si le responsable du paquet pense que la plupart des utilisateurs ne voudraient pas A sans avoir également la fonctionnalité fournie par B.

— Le paquet A *suggère* le paquet B si B contient des fichiers qui sont liés aux (et augmentent habituellement) fonctionnalités de A.

— Le paquet A est en *conflit* avec le paquet B quand A ne peut pas fonctionner si B est installé sur le système. Le plus souvent les conflits sont des cas où A contient des fichiers qui sont une amélioration de ceux de B. Le conflit est souvent combiné avec le remplacement.

— Le paquet A *remplace* le paquet B quand les fichiers installés par B sont supprimés et (dans certains cas) écrasés par les fichiers de A.

— Le paquet A *casse* le paquet B quand les deux paquets ne peuvent pas être simultanément configurés. Le système de gestion des paquets refusera d'installer l'un des paquets si l'autre est déjà installé et configuré dans le système.

— Le paquet A *fournit* le paquet B quand tous les fichiers et toutes les fonctionnalités de B sont incorporés dans A. Ce mécanisme fournit aux utilisateurs ayant une contrainte d'espace disque un moyen d'obtenir seulement la partie du paquet dont ils ont vraiment besoin.

Une information plus détaillée sur l'utilisation de chacun de ces termes peut être trouvée dans la charte Debian, consultez le chapitre 7.2 « Dépendances binaires ».

7.10 Quel est le sens de « Pre-Depends » ?

« Pre-Depends » est une dépendance particulière. Dans le cas de la plupart des paquets, `dpkg` désarchivera les fichiers archives (c'est-à-dire les fichiers `.deb`) indépendamment de l'existence sur le système des fichiers dont il dépend. De manière simpliste, le désarchivage signifie que `dpkg` extraira à partir du fichier archive les fichiers qui sont censés être installés sur votre système de fichiers, et les mettra en place. Si ces paquets dépendent de l'existence de quelques autres paquets sur votre système, `dpkg` refusera d'accomplir l'installation (par l'exécution de l'action de configuration) jusqu'à ce que les autres paquets soient installés.

Cependant, pour certain paquets, `dpkg` refusera de les désarchiver jusqu'à ce que certaines dépendances soient résolues. De tels paquets sont dits « Pré-dépendants » de la présence de quelques autres paquets. Le projet Debian fournissait ce mécanisme pour gérer la mise à jour de systèmes utilisant le format `a.out` vers le format `ELF`, où l'ordre de désarchivage était critique. Il y a d'autres situations de mise à jour où cette méthode est utile, par exemple pour les paquets ayant une priorité « required » et des dépendances sur la libc.

Comme ci-dessus, une information plus détaillée sur l'utilisation de ce terme peut être trouvée dans la charte Debian.

7.11 Quelle est la signification de *unknown*, *install*, *remove*, *purge* et *hold* dans le statut du paquet ?

Ces drapeaux indiquent ce que souhaite faire l'utilisateur avec un paquet (ils sont positionnés par les actions de l'utilisateur dans la section « Section » de `dselect` ou par des appels directs de `dpkg` par l'utilisateur).

Leurs significations sont :

— unknown - L'utilisateur n'a jamais indiqué s'il souhaitait le paquet.

— install - L'utilisateur souhaite installer ou mettre à jour le paquet.

— remove - L'utilisateur souhaite supprimer le paquet, mais ne désire pas effacer les fichiers de configuration existants.

— purge - L'utilisateur souhaite supprimer le paquet complètement, incluant ses fichiers de configuration.

— hold - L'utilisateur ne souhaite pas traiter le paquet, c.-à-d. qu'il veut conserver la version courante avec l'état actuel de celui-ci.

7.12 Comment est-ce que je mets un paquet en « hold » ?

Il y a trois moyens de bloquer un paquet, avec dpkg, aptitude ou dselect.

Avec dpkg, vous devez exporter la liste de sélection des paquets, avec la commande :

```
dpkg --get-selections \* > selections.txt
```

puis éditer le fichier `selections.txt`, changer la ligne contenant le paquet que vous souhaitez conserver, par exemple pour `libc6`, de ceci :

```
libc6                                           install
```

à ceci :

```
libc6                                           hold
```

sauvegarder le fichier et le recharger dans la base de données de dpkg avec :

```
dpkg --set-selections < selections.txt
```

Avec aptitude, vous pouvez conserver un paquet en faisant

```
aptitude hold package_name
```

et enlever le drapeau « hold » avec

```
aptitude unhold package_name
```

Avec dselect, vous devez entrer dans l'écran [S]elect, trouver le paquet que vous souhaitez conserver dans l'état actuel, et appuyer sur la touche '=' (ou sur 'H'). Les changements seront pris en compte à la sortie de l'écran [S]elect.

7.13 Comment est-ce que j'installe un paquet source ?

Les paquets sources Debian ne peuvent pas être véritablement installés, ils sont seulement dépaquetés dans n'importe quel répertoire où vous voulez en construire les paquets binaires.

Les paquets sources sont distribués sur la plupart des serveurs où vous pouvez obtenir les paquets binaires. Si vous configurez votre fichier `sources.list(5)` pour APT en ajoutant la ligne « deb-src » appropriée, vous pourrez facilement télécharger toutes les sources des paquets en lançant la commande

```
apt-get source foo
```

Pour vous aider réellement à la construction du paquet de source, Debian fournit un mécanisme pour les dépendances de construction. Ceci signifie que le responsable du paquet tient une liste des autres paquets nécessaires à la construction du paquet. Pour voir comment cela fonctionne, exécutez

```
apt-get build-dep foo
```

avant de construire les sources.

7.14 Comment est-ce que je peux construire les paquets binaires depuis les paquets sources ?

La méthode recommandée est d'utiliser les différents outils de haut-niveau (« wrappers »). Voici comment faire en utilisant les outils `devscripts`. Installez le paquet si cela n'est pas déjà fait.

Maintenant, commencez par récupérer le paquet source :

```
apt-get source foo
```

puis entrez dans l'arborescence source :

```
cd foo-*
```

Puis installez les dépendances de construction nécessaires (s'il y en a) :

```
sudo apt-get build-dep foo
```

Puis créez une version particulière de votre propre construction (afin de ne pas les confondre plus tard lorsque Debian publiera sa propre nouvelle version) :

```
dch -l local 'Blah blah blah'
```

Et enfin construisez votre paquet :

```
debuild -us -uc
```

Si tout a correctement fonctionné, vous deviez pouvoir installer votre paquet avec la commande :

```
sudo dpkg -i ../*.deb
```

Si vous préférez effectuer les opérations vous-même, et ne voulez pas utiliser `devscripts`, suivez cette procédure :

Vous aurez besoin de tous les fichiers foo_*.dsc, foo_*.tar.gz et foo_*.diff.gz pour compiler les sources (remarquez que pour certains paquets il n'y a pas de fichier .diff.gz, ce sont les paquets natifs à Debian).

Une fois que vous les avez récupérés (voir 'Comment est-ce que j'installe un paquet source ?' page ci-contre), si vous avez le paquet `dpkg-dev` installé, la commande suivante :

```
dpkg-source -x foo_version-revision.dsc
```

désarchivera le paquet dans un répertoire nommé `foo-version`.

Si vous souhaitez juste compiler le paquet, vous devez vous placer dans le répertoire `foo-version` et lancer la commande

```
dpkg-buildpackage -rfakeroot -b
```

pour construire le paquet (notez que la paquet `fakeroot` est aussi nécessaire) et faites

```
dpkg -i ../foo_version-revision_arch.deb
```

pour installer le nouveau paquet.

7.15 Comment est-ce que je crée un paquet Debian ?

Pour une description plus détaillée, lisez le « guide des nouveaux responsables Debian », disponible dans le paquet `maint-guide` ou à l'adresse `http://www.debian.org/doc/devel-manuals#maint-guide`.

Chapitre 8

Outils de gestion des paquets Debian

8.1 Quels sont les programmes fournis par Debian pour la gestion des paquets ?

Il y a beaucoup d'outils utilisés pour la gestion des paquets Debian, depuis des outils basés sur des interfaces texte ou graphique aux outils de bas niveau utilisés pour l'installation des paquets. Tous les outils disponibles reposent sur les outils de plus bas niveau pour fonctionner correctement et sont présentés ici selon un niveau de complexité décroissant.

Il est important de comprendre que les outils de gestion des paquets Debian de haut niveau comme `aptitude` ou `dselect` reposent sur `apt` qui lui-même utilise `dpkg` pour la gestion des paquets sur le système.

Consultez le Chapitre 2. Gestion des paquets Debian (`http://www.debian.org/doc/manuals/debian-reference/ch02`) de la Référence Debian (`http://www.debian.org/doc/manuals/debian-reference/`) pour plus d'informations sur les outils de gestion des paquets Debian. Ce document est disponible dans différentes langues et formats (`http://www.debian.org/doc/user-manuals#quick-reference`).

8.1.1 dpkg

C'est le programme principal de gestion des paquets. `Dpkg` peut être exécuté avec beaucoup d'options. Quelques utilisations courantes :

— Afficher toutes les options : `dpkg --help`.

— Afficher le fichier control (et d'autres informations) pour un paquet particulier : `dpkg --info foo_VVV-RRR.deb`

— Installer un paquet (incluant le désarchivage et la configuration) sur le disque dur : `dpkg --install foo_VVV-RRR.deb`.

— Désarchiver (mais sans configurer) une archive Debian dans le système de fichiers du disque dur : `dpkg --unpack foo_VVV-RRR.deb`. Notez que cette opération ne laisse *pas* nécessairement le paquet dans un état utilisable ; certains fichiers peuvent avoir besoin de personnalisation pour fonctionner correctement. Cette commande supprime toute version déjà installée du programme et exécute le script preinst (voir 'Qu'est-ce qu'un script Debian preinst, postinst, prerm, et postrm ?' page 29) du paquet.

— Configurer un paquet qui a déjà été désarchivé : `dpkg --configure foo`. Entre autre chose, cette action lance le script postinst (voir 'Qu'est-ce qu'un script Debian preinst, postinst, prerm, et postrm ?' page 29) du paquet. Il met aussi à jour les fichiers listés dans le fichier `conffiles` pour ce paquet. Notez que l'option configure prend un nom de paquet (par exemple, foo) et *non* le nom de l'archive Debian (par exemple, foo_VVV-RRR.deb).

— Extraire un fichier nommé « blurf » (ou un groupe de fichiers nommés « blurf* ») depuis une archive Debian : `dpkg --fsys-tarfile foo_VVV-RRR.deb | tar -xf - blurf*`

— Supprimer un paquet (mais pas ses fichiers de configuration) : `dpkg --remove foo`.

— Supprimer un paquet (ainsi que ses fichiers de configuration) : `dpkg --purge foo`.

— Lister l'état de l'installation des paquets contenant la chaîne (ou l'expression rationnelle) « foo* » : `dpkg --list 'foo*'`.

8.1.2 APT

APT est le *Advanced Package Tool* et fournit le programmme `apt-get`. `Apt-get` fournit un moyen simple pour installer des paquets depuis la ligne de commande. À la différence de `dpkg`, `apt-get` ne comprend pas les fichiers .deb, il travaille avec

le nom du paquet et peut seulement installer les archives .deb depuis une source indiqué dans `/etc/apt/sources.list`. `Apt-get` peut appeler directement `dpkg` après avoir téléchargé l'archive [1] .deb depuis les sources.

Les options les plus courantes de `apt-get` :

— Pour mettre à jour la liste des paquets connus par votre système :

```
apt-get update
```

(vous devriez exécuter cette commande régulièrement pour mettre à jour vos listes de paquets)

— Pour mettre à jour tous les paquets de votre système, sans installer de paquets supplémentaires ou en supprimer :

```
apt-get upgrade
```

— Pour installer le paquet *foo* et toutes ses dépendances :

```
apt-get install foo
```

— Pour supprimer le paquet `foo` de votre système :

```
apt-get remove foo
```

— Pour supprimer le paquet `foo` et ses fichiers de configuration de votre système :

```
apt-get --purge remove foo
```

— Pour mettre à jour votre système entier, en permettant si nécessaire l'installation de paquets supplémentaires ou la suppression de paquets :

```
apt-get dist-upgrade
```

(La commande `upgrade` conserve les paquets dans leur version actuelle si la mise à jour nécessite l'installation de paquets supplémentaires pour satisfaire une nouvelle dépendance. La commande `dist-upgrade` est moins stricte.)

Notez que vous devez être authentifié en tant que root pour exécuter toutes commandes qui modifient le système de paquets.

Notez que `apt-get` installe désormais par défaut les paquets recommandés et constitue le programme de référence pour la gestion des paquets en console, leur installation mais aussi la mise à jour du système.

La suite d'outils apt inclut aussi le programme `apt-cache` pour questionner les listes de paquet. Vous pouvez l'utiliser pour trouver des paquets fournissant des fonctionnalités spécifiques à l'aide d'une simple requête ou d'une expression rationnelle et par des questions sur les dépendances dans le système de gestion des paquets. Les options les plus courantes de `apt-cache` :

— Pour trouver les paquets dont la description contient *mot* :

```
apt-cache search mot
```

— Pour afficher des informations détaillées sur un paquet :

```
apt-cache show paquet
```

— Pour afficher les dépendances d'un paquet :

```
apt-cache depends paquet
```

— Pour afficher des informations détaillées des versions disponibles pour un paquet et les paquets ayant des dépendances inverses sur lui :

```
apt-cache showpkg paquet
```

Pour plus d'informations, installez le paquet `apt` et lisez les pages de manuel `apt-get(8)` et `sources.list(5)`. Vous pouvez aussi lire la documentation sur apt-get, installez le paquet `apt-doc` et lisez `/usr/share/doc/apt-doc/guide.html/index.html`.

1. Notez qu'il y a des portages qui rendent cet outil disponible pour d'autres systèmes de gestion de paquet, comme le gestionnaire de paquets de Red Hat, aussi connu sous `rpm`

8.1.3 aptitude

`Aptitude` est un gestionnaire de paquet pour les systèmes Debian GNU/Linux qui fournit une interface au gestionnaire de paquet apt. `Aptitude` est une interface texte basée sur la bibliothèque curses, il peut être utilisé pour effectuer des tâches d'administration de manière rapide et facile.

`Aptitude` fournit les mêmes fonctionnalités que `dselect` et `apt-get`, ainsi que des fonctions additionelles non présentes dans les autres programmes :

— `Aptitude` permet d'accéder facilement à toutes les versions d'un paquet.

— `Aptitude` facilite la recherche de logiciels obsolètes en les listant sous « Paquets obsolètes ou créés localement »

— `Aptitude` inclut un système assez puissant pour rechercher des paquets particuliers. Les utilisateurs familiers avec `mutt` le maîtriseront rapidement car la syntaxe a été inspirée de celle de `mutt`.

— `Aptitude` peut être utilisé pour installer les tâches prédéfinies. Pour plus d'information, consultez 'tasksel' de la présente page.

— `Aptitude` en mode plein écran intègre les fonctionnalités de `su`, et peut être utilisé par un utilisateur normal. Le programme `su` sera appelé (et le mot de passe root demandé) lorsque les privilèges d'administrateur seront nécessaires.

Vous pouvez utiliser `aptitude` au travers d'une interface graphique (en lançant simplement `aptitude`) ou directement en ligne de commande. La syntaxe utilisée en ligne de commande est très similaire à celle utilisée avec `apt-get`. Par exemple, pour installer le paquet *foo* vous pouvez lancer `aptitude install` *`foo`*.

Notez que `aptitude` est le gestionnaire de paquet recommandé pour la gestion quotidienne des paquets en console.

Pour plus d'informations, lisez la page de manuel `aptitude(8)` et installez le paquet de documentation `aptitude-doc`.

8.1.4 synaptic

`synaptic` est un gestionnaire graphique de paquets qui vous permet d'installer, de mettre à jour et de supprimer des paquets logiciels d'une manière conviviale. En plus des fonctionnalités offertes par aptitude, `synaptic` permet d'éditer la liste des dépôts de logiciels et de parcourir toute la documentation relative à un paquet. Pour plus d'informations, consulez le site web de Synaptic (http://www.nongnu.org/synaptic/).

8.1.5 tasksel

Lorsque vous désirez effectuer une tâche particulière, il peut être difficile de trouver les paquets appropriés à votre besoin. Les développeurs Debian ont défini des `tâches` : une tâche est un ensemble de paquets Debian distincts qui sont tous destinés à la même activité. Les tâches peuvent être installées en utilisant le programme `tasksel` ou `aptitude`.

Typiquement, l'installateur Debian installe automatiquement la tâche associée aux systèmes standards ainsi que la tâche d'environnement bureautique. L'environnement bureautique installé dépendra du support CD/DVD utilisé, il s'agira la plupart du temps du bureau GNOME (tâche `gnome-desktop`). Ainsi, en fonction de vos choix lors de l'installation, des tâches seront automatiquement installées sur votre système. Par exemple, la tâche correspondante à votre langue sera installée, ainsi que la tâche `laptop` si vous utilisez un ordinateur portable reconnu comme tel par l'installateur.

8.1.6 Autres outils de gestion des paquets

dselect

Ce programme est une interface pilotée par menu pour le système de gestion de paquet de Debian. Pour Woody et les publications précédentes, il s'agissait de la principale interface de gestion des paquets. Aujourd'hui, il est reommandé aux utilisateurs de se tourner plutôt vers `aptitude`, outil avec lequel les utilisateurs se sentent plus à l'aise. De plus `aptitude` est plus recommandé que `dselect` pour les mises à niveau à grande échelle. Pour plus d'informations sur `aptitude`, veuillez consultez 'aptitude' de la présente page.

`dselect` peut :

— guider l'utilisateur dans son choix parmi les paquets à installer ou à enlever, en s'assurant qu'on n'installe aucun paquet qui soit en conflit entre eux et que tous les paquets nécessaires au fonctionnement correct de chaque paquet sont installés ;

— avertir l’utilisateur sur les contradictions ou les incompatibilités de ses choix ;
— déterminer l’ordre d’installation des paquets ;
— effectuer automatiquement l’installation ou la suppression et
— guider l’utilisateur à travers le processus de configuration exigé pour chaque paquet.

Au lancement `dselect` présente à l’utilisateur un menu avec sept choix, chacun d’eux est une action spécifique. L’utilisateur peut choisir l’une de ces actions, en déplaçant la sélection avec les flèches du clavier, puis en appuyant sur la touche <*entrée*> pour choisir l’action désirée.

Ce que l’utilisateur voit va dépendre de l’action qu’il a choisi. S’il choisit une des options autre que `Accéder` ou `Sélection`, alors `dselect` exécutera simplement l’action choisie : par exemple, si l’utilisateur choisit l’action `Retirer`, alors `dselect` procédera à la suppression de tous les fichiers marqués à supprimer la dernière fois que l’utilisateur a choisi l’action `Sélection`.

Les deux menus, `Accéder` et `Sélection`, amènent dans un nouveau menu. Dans les deux cas, l’écran est divisé en deux : l’écran du haut donne une liste défilable de choix, tandis que l’écran du bas donne une courte explication de chaque choix.

Une aide en ligne est disponible, appuyez sur la touche < ?> pour obtenir l’aide à tout moment.

L’ordre de présentation des actions dans le premier menu de `dselect` reflète l’ordre dans lequel un utilisateur exécutera normalement `dselect` pour installer des paquets. Cependant, un utilisateur peut sélectionner n’importe lequel des choix principaux de menu aussi souvent que nécessaire (y compris en ignorer, selon ce qu’il veut faire).

— Premièrement choisir la **Méthode d’accès**. C’est la méthode par laquelle l’utilisateur projette d’accéder aux paquets de Debian ; par exemple, certains utilisateurs accèdent aux paquets Debian sur cédéroms tandis que d’autres y accèderont par FTP anonyme. La « méthode d’accès » est conservée après la sortie de `dselect` donc si elle ne change pas, cette option n’a plus besoin d’être choisie.

— Ensuite **Mise à jour** de la liste des paquets disponibles. Pour faire cela, `dselect` lit le fichier « Packages.gz » qui devrait être inclus à la racine du répertoire où les paquets Debian installés sont conservés. (Mais s’il ne le trouve pas, `dselect` vous proposera de le créer pour vous.)

— **Sélection** des paquets spécifiques pour installation sur le système. Après avoir choisi ce menu, l’utilisateur se retrouve face à un écran d’aide (à moins que l’option « –expert » a été utilisée). Une fois que l’utilisateur a quitté l’écran d’aide, il lui est présenté un écran divisé en deux parties, pour choisir les paquets à installer (ou à supprimer).

  La partie supérieure de l’écran est une fenêtre relativement étroite montrant la liste des 37400 paquets Debian ; dans la partie inférieure de l’écran est affichée la description du paquet ou des groupes de paquets qui ont été sélectionnés au dessus.

  Vous pouvez indiquer quels paquets devraient être activés en mettant en surbrillance le nom d’un paquet ou d’un groupe de paquets. Après quoi, vous pouvez sélectionner des paquets :

  pour être installés : Ceci est accompli en appuyant sur la touche ’+’.

  pour être supprimés : Les paquets peuvent être effacés de deux façons :

    — supprimé : ceci supprime la plupart des fichiers associés avec le paquet, mais conserve les fichiers listés comme fichiers de configuration (voir ‘Qu’est ce qu’un Debian conffile ?’ page 29) et l’information de configuration du paquet. Ceci est accompli en appuyant sur la touche ’-’.

    — purgé : ceci supprime *tous* les fichiers qui font partie du paquet. Ceci est accompli en appuyant sur la touche ’_’.

    Notez qu’il n’est pas possible de supprimer tous les paquets. Si vous essayez de faire cela, votre système sera à la place réduit aux paquets de base installés initialement.

  pour les bloquer (« on hold ») : Ceci est accompli en appuyant sur la touche ’=’, ceci indique à `dselect` de ne pas mettre à jour le paquet même si la version installée sur votre système n’est pas la version la plus récente qui est disponible dans le référentiel Debian que vous utilisez (défini quand vous avez indiqué la « méthode d’accès » et obtenu quand vous avez mis à jour la liste de paquet disponible).

    Vous pouvez inverser la configuration en appuyant sur la touche ’ :’. ceci indique à `dselect` que le(s) paquet(s) peut(peuvent) de nouveau être mis à jour s’il existe une version disponible. C’est la configuration par défaut.

  On peut changer l’ordre de présentation en utilisant la touche ’o’ qui va le changer cycliquement. L’ordre par défaut présente les paquets par priorité ; pour chaque priorité, les paquets sont présentés selon le répertoire (c.-à-d. par section) des archives dans lesquelles ils sont stockés. De cette manière, quelques paquets (disons) de la section A peuvent apparaître en premier, suivis de paquets de la section B, suivis d’autres paquets (de priorité plus faible) de la section A.

Vous pouvez aussi agrandir les étiquettes de la partie supérieur de l'écran en utilisant la touche 'v' (verbeux). Cette action déplace une grande partie du texte qui était adapté à l'affichage vers la droite de l'écran. Pour le faire défiler, utilisez les flèches droite et gauche du clavier.

Si vous choisissez un paquet à installer ou à supprimer, par exemple, le `foo.deb` et que ce paquet dépend (ou recommande) un autre paquet, par exemple `blurf.deb`, alors `dselect` vous présentera un sous-écran de l'écran principal de sélection. Là vous pourrez choisir parmi les paquets relatifs, en acceptant les actions suggérées (pour installer ou pas), ou en les rejetant. Pour le faire plus tard, appuyez sur Maj-D ; pour retourner aux suggestions appuyez sur Maj-U. Dans tous les cas, vous pouvez sauvegarder votre sélection et retourner au menu principal de sélection en appuyant sur Maj-Q.

— Une fois revenu au menu principal les utilisateurs peuvent choisir le menu « Installer » pour désarchiver et configurer les paquets choisis. Alternativement, les utilisateurs qui souhaitent supprimer des fichiers peuvent choisir le menu « Retirer » n'importe où. L'utilisateur peut choisir l'option « Quitter » pour sortir de `dselect` ; toutes les sélections seront sauvegardées par `dselect`.

dpkg-deb

Ce programme travaille sur les fichiers archive Debian (`.deb`). Quelques utilisations courantes :

— Afficher toutes les options : `dpkg-deb --help`.
— Afficher les fichiers présents dans une archive Debian : `dpkg-deb --contents foo_VVV-RRR.deb`)
— Extraire les fichiers contenus dans une archive Debian dans un répertoire particulier de l'utilisateur : `dpkg-deb --extract foo_VVV-RRR.deb tmp` extrait tous les fichiers de `foo_VVV-RRR.deb` dans le répertoire `tmp/`. C'est pratique pour examiner le contenu d'un paquet dans un répertoire localisé, sans installer le paquet sur le système de fichiers racine.
— Extraire les fichiers de contrôle du paquet : `dpkg-deb --control foo_VVV-RRR.deb tmp`.

Notez que tous les paquets qui ont été simplement désarchivés en utilisant `dpkg-deb --extract` ne seront pas correctement installés, vous devriez employer `dpkg --install` à la place.

Vous trouverez plus d'informations dans la page de manuel `dpkg-deb(1)`.

dpkg-split

Ce programme permet de diviser de gros paquets en fichiers plus petits (par exemple, pour les copier sur un ensemble de disquettes) et peut aussi être utilisé pour concaténer un ensemble de fichiers dans un seul fichier. On peut seulement l'utiliser sur un système Debian (c.-à-d. un système contenant le paquet `dpkg`), puisqu'il utilise le programme `dpkg-deb` pour analyser le paquet Debian.

Par exemple, pour diviser un gros fichier .deb en N parties,

— Exécuter la commande `dpkg-split --split foo.deb`. Ceci créera N fichiers d'une longueur approximative de 460 Ko chacun dans le répertoire courant.
— Copier ces fichiers sur des disquettes.
— Copier le contenu des disquettes sur le disque dur de votre choix sur une autre machine.
— Concaténer ces fichiers ensemble en utilisant `dpkg-split --join "foo*"`.

8.2 Debian prétend pouvoir mettre à jour un programme en fonctionnement ; comment est-ce que c'est possible ?

Le noyau (et le système de fichiers) dans les système Debian GNU/Linux permet le remplacement de fichiers même lorsqu'ils sont utilisés.

Debian fournit aussi un programme appelé `start-stop-daemon` qui est utilisé pour activer les démons au démarrage du système ou les arrêter lors d'un changement de niveau d'execution (par exemple, lors du passage du mode multi-utilisateur au niveau mono-utilisateur ou lors de l'arrêt du système). Ce programme est aussi utilisé par les scripts d'installation, lors de l'installation d'un paquet contenant un démon, pour arrêter les démons en fonctionnement et les redémarrer si nécessaire.

8.3 Comment savoir si un paquet est déjà installé sur un système Debian ?

Pour connaître l'état de tous les paquets installés sur un système Debian exécutez la commande

```
dpkg --list
```

Ceci affiche un résumé sur une ligne pour chaque paquet, donnant un symbole de l'état avec deux lettres (expliqué dans l'entête), le nom du paquet, la version *installée*, et une courte description.

Vous pouvez connaître l'état de paquets dont le nom correspond au motif commençant par « foo » en exécutant la commande :

```
dpkg --list 'foo*'
```

Pour obtenir un rapport plus détaillé pour un paquet particulier, lancez la commande :

```
dpkg --status packagename
```

8.4 Comment afficher la liste des fichiers d'un paquet installé ?

Pour lister tous les fichiers fournis par le paquet installé `foo`, exécutez la commande suivante :

```
dpkg --listfiles foo
```

Rappelez-vous que les fichiers créés par les scripts d'installation ne sont pas affichés.

8.5 Comment trouver le paquet contenant un fichier particulier ?

Pour identifier le paquet qui a produit le fichier appelé `foo`, exécutez l'une des commandes suivantes :

— `dpkg --search foo`

Ceci recherche `foo` dans les paquets installés. (Ceci est équivalent à la recherche de tous les fichiers ayant une extension `.list` dans le répertoire `/var/lib/dpkg/info/` et à afficher le nom de tous les paquets le contenant).

Une alternative plus rapide à ceci est l'outil `dlocate`.

```
dlocate -S  filename
```

— `zgrep foo Contents-ARCH.gz`

Ceci recherche les fichiers contenant la sous-chaîne `foo` dans leurs noms de chemin complet. Les fichiers `Contents-ARCH.gz` (où ARCH représente l'architecture voulue) se trouvent dans les répertoires de paquets principaux (main, non-free, contrib) sur un site FTP Debian (c.-à-d. sous `/debian/dists/wheezy`). Un fichier `Contents` se réfère seulement aux paquets contenus dans le sous-répertoire où il se trouve. Par conséquent, un utilisateur doit rechercher dans plus d'un fichier `Contents` pour trouver le paquet contenant le fichier `foo`.

Cette méthode a l'avantage par rapport à `dpkg --search` parce qu'elle permet de trouver des fichiers dans des paquets qui ne sont pas installés sur votre système.

— `apt-file search` *`foo`*

Si vous installez `apt-file`, similaire à ce qui précède, recherche des fichiers qui contiennent la sous-chaîne ou l'expression rationelle `foo` dans leurs noms de chemin. L'avantage par rapport à ci-dessus est qu'il n'y a aucun besoin de rechercher les fichiers `Contents-ARCH.gz` car ceci est fait automatiquement pour toutes les sources définies dans le fichier `/etc/apt/sources.list` quand vous lancez (sous le compte root) `apt-file update`.

8.6 Pourquoi « foo-data » n'est-il pas supprimé quand je désinstalle « foo » ? Comment m'assurer que des paquets obsolètes de bibliothèques sont purgés ?

Certains paquets sont découpés en un paquet pour le programme (« foo ») et un paquet pour les données (« foo-data ») (ou en « foo » et « foo-doc »). Cela est vrai pour beaucoup de jeux, d'applications multimédia et de dictionnaires dans Debian. Cela a été introduit parce que certains utilisateurs pourraient vouloir accéder aux données sans installer le programme ou parce que le programme peut être exécuté sans les données.

Des situations similaires apparaissent lorsqu'il s'agit de gérer les bibliothèques : généralement elles sont installées puisque des paquets applicatifs dépendent d'elles. Lorsqu'un paquet applicatif est purgé, le paquet fournissant la bibliothèque peut être conservé sur le système. Par exemple, lorsqu'une application ne dépend plus de libdb4.2, mais de libdb4.3, le paquet libdb4.2 peut être conservé après la mise à jour du paquet applicatif.

Dans ces cas, « foo-data » ne dépend pas de « foo », donc quand vous supprimez le paquet « foo » la plupart des outils de gestion de paquets ne supprimeront pas automatiquement « foo-data ». Cela reste valable pour les paquets de bibliothèques. Ceci est nécessaire pour ne pas avoir de dépendance circulaire. Si vous utilisez `aptitude` (voir 'aptitude' page 37) comme outil de gestion de paquet, il dépistera les paquets installés automatiquement et les supprimera quand ils ne seront plus utilisés par aucun paquet.

Chapitre 9

Garder son système Debian à jour

L'un des buts de Debian est de fournir un chemin de mise à jour compatible et un processus de mise à jour sûr. Nous faisons toujours de notre mieux pour que la mise à jour des nouvelles versions soit une procédure douce. Dans certain cas, il peut y avoir des avertissements importants lors du processus de mise à jour, l'utilisateur en sera alerté et souvent une solution à un problème possible sera fournie.

Vous pouvez aussi lire les notes de publication, qui décrivent en détail les spécificités de la mise à jour, disponibles en ligne à l'adresse http://www.debian.org/releases/stable/releasenotes et présentes sur tous les CD, DVD et Blu-Ray de Debian.

9.1 Comment puis-je garder mon système à jour ?

On pourrait simplement ouvrir une session FTP anonyme vers une archive Debian, parcourir les répertoires jusqu'à ce qu'on trouve le fichier désiré, le récupérer et enfin l'installer en utilisant `dpkg`. Notez que dpkg installera les fichiers mis à jour à leur place, même sur un système en marche. Parfois, la mise à jour d'un paquet aura besoin de l'installation d'une nouvelle version d'un autre paquet, auquel cas l'installation échouera si l'autre paquet n'est pas installé.

Beaucoup de gens trouvent cette approche trop gourmande en temps, car Debian évolue très rapidement ; typiquement, une douzaine ou plus de nouveaux paquets sont téléchargés chaque semaine. Ce nombre est encore plus grand avant la sortie d'une version majeure. Pour gérer cette avalanche, beaucoup de gens préfèrent utiliser une méthode automatique. Plusieurs outils de gestion des paquets sont disponibles dans ce but :

9.1.1 aptitude

APT est une interface avancée pour le système de gestion des paquets Debian. Il fournit les fonctionnalités d'ordonnancement d'installation complet, de possibilités de sources multiples et plusieurs autres fonctionnalités uniques, voir le guide de l'utilisateur `/usr/share/doc/apt-doc/guide.html/index.html` (vous pouvez aussi installer le paquet `apt-doc`).

`aptitude` est le gestionnaire de paquets recommandé pour les systèmes Debian GNU/Linux. C'est une interface en mode texte à APT qui utilise la bibliothèque curses et peut être utilisé pour améliorer la gestion des tâches de façon rapide et facile.

Avant de pouvoir utiliser `aptitude`, vous devrez éditer le fichier `/etc/apt/sources.list` pour le configurer. Si vous souhaitez mettre à jour votre système vers la dernière version stable de Debian, vous voudrez probablement utiliser une ligne comme celle-là :

```
http://ftp.us.debian.org/debian stable main contrib non-free
```

Vous pouvez remplacer ftp.us.debian.org avec le nom d'un miroir Debian plus proche de chez vous et plus rapide. Voir la liste des miroirs à l'adresse http://www.debian.org/mirror/list pour plus d'informations.

Vous trouverez plus d'informations en lisant la pages de manuel `sources.list(8)`.

Pour mettre à jour votre système, exécutez

```
aptitude update
```

suivi de

```
aptitude dist-upgrade
```

Répondez à toutes les questions qui vous seront posées et votre système sera mis à jour. Voir aussi 'aptitude' page 37.

9.1.2 apt-get, dselect et apt-cdrom

`apt-get` est un outil en ligne de commandes basé sur APT pour gérer les paquets et la méthode APT dselect est une interface pour APT par `dselect`. Les deux sont un moyen simple et sûr d'installer et de mettre à jour des paquets.

Pour utiliser `apt-get`, installez le paquet `apt`, et éditez le fichier `/etc/apt/sources.list` pour le configurer, tout comme pour 'aptitude' page précédente.

Exécutez

```
apt-get update
```

suivi de

```
apt-get dist-upgrade
```

Répondez à toutes les questions qui vous seront posées et votre système sera mis à jour. Vous trouverez plus d'informations en lisant la pages de manuel `apt-get(8)` et aussi 'APT' page 35.

Pour utiliser APT avec `dselect`, choisissez la méthode d'accès par APT dans l'écran de sélection de méthode de dselect (option 0) et indiquez les sources devant être utilisées. Le fichier de configuration est `/etc/apt/sources.list`. Vous pouvez également consulter 'dselect' page 37.

Si vous souhaitez utiliser les CD-ROM pour installer les paquets, vous pouvez utiliser le programme `apt-cdrom`. Pour plus de détails, veuillez-vous reporter aux notes de publication, paragraphe « Ajouter des sources Internet à apt ».

Veuillez noter que quand vous installez un paquet, les archives sont conservées dans un sous-répertoire de /var. Pour ne pas saturer votre partition, vous devriez supprimer les archives en utilisant `apt-get clean` et `apt-get autoclean` ou les déplacer à un autre endroit (conseil : utiliser `apt-move`).

9.1.3 aptitude

`aptitude` est une interface en mode texte pour le système de paquets Debian. Elle permet à l'utilisateur de voir la liste des paquets et d'effectuer des tâches de gestion des paquets telles que l'installation, la mise à jour ou la suppression de paquets (consultez 'aptitude' page 37). Les actions peuvent être effectuées à partir d'une interface visuelle ou en ligne de commandes.

En ligne de commandes, les actions sont similaires à celles de APT ; pour mettre à jour votre système, exécutez

```
aptitude update
```

suivi de

```
aptitude dist-upgrade
```

Notez qu'`aptitude` n'est plus l'outil recommandé pour mettre à niveau d'une version de Debian GNU/Linux à une autre. La méthode de mise à niveau entre versions est décrite dans les url id="http ://www.debian.org/releases/stable/releasenotes" name="notes de publication">. Ce document décrit en détail les étapes recommandées ainsi que les problèmes à connaître avant la mise à jour.

Vous trouverez plus d'informations en lisant la pages de manuel `sources.list(8)` et le fichier `/usr/share/aptitude/README`.

9.1.4 mirror

Ce script Perl et son programme (optionel) de gestion nommé `mirror-master`, peuvent être employés pour rechercher des parties indiquées par l'utilisateur d'une arborescence de répertoire sur un serveur particulier *via* un FTP anonyme.

`Mirror` est particulièrement utile pour télécharger un grand nombre de logiciels. Après le premier téléchargement, un fichier nommé `.mirrorinfo` est conservé sur l'ordinateur local. Les changements du système de fichiers distant sont détectés automatiquement par `mirror`, qui compare le fichier local avec un fichier semblable sur le système distant et télécharge seulement les fichiers modifiés.

Le programme `mirror` est généralement utile pour mettre à jour les copies locales des arborescences de répertoire distant. Les fichiers cherchés n'ont pas besoin d'être des fichiers Debian. (Depuis que le programmme `mirror` a été porté en Perl, il peut aussi être exécuté sur des systèmes non-Unix). Bien que le programme `mirror` fournisse des mécanismes pour exclure des fichiers dont les noms correspondent à des chaînes indiquées par l'utilisateur, ce programme est plus utile quand son objectif est de télécharger des arborescences entières, plutôt que des paquets choisis.

9.1.5 dpkg-mountable

Dpkg-mountable ajoute une méthode d'accès nommée « mountable » à la liste de dselect, qui vous permet d'installer depuis tout système de fichiers indiqué dans /etc/fstab. Par exemple, l'archive peut être sur un disque dur local ou sur un serveur NFS qui sera automatiquement monté et démonté si nécessaire.

Il a aussi quelques fonctionnalités supplémentaires que l'on ne trouve pas avec les méthodes standards de dselect, telle que la disposition dans une arborescence locale de fichier (soit en parallèle de la distribution principale soit totalement séparée), et en obtenant seulement les paquets qui sont exigés, plutôt qu'en effectuant un long balayage récursif de répertoires et également la journalisation de toutes les actions de dpkg dans la méthode d'installation.

9.2 Faut-il être dans le mode mono-utilisateur pour mettre à jour un paquet ?

Non. Les paquets peuvent être mis à jour en mode multi-utilisateur, même quand le système est en fonctionement. Debian fournit le programme `start-stop-daemon` qui permet d'arrêter et de redémarrer les processus en cours d'exécution si cela est nécessaire pendant la mise à jour du paquet.

9.3 Faut-il garder toutes les achives .deb sur le disque ?

Non. Si vous avez téléchargé les fichiers sur votre disque, alors après avoir installé les paquets vous pouvez les supprimer de votre système, par exemple en exécutant `aptitude clean`.

9.4 Comment puis-je garder un journal des paquets que j'ai ajoutés au système ? J'aimerais savoir quand et quelles mises à jour et suppressions de paquets ont eu lieu !

Passer l'option `--log` à `dpkg` active le journal des mises à jour et suppression de `dpkg`. Cela enregistre à la fois l'appel de `dpkg`-invokation (e.g.

```
2005-12-30 18:10:33 install hello 1.3.18 2.1.1-4
```

) et les résultats (par exemple

```
2005-12-30 18:10:35 status installed hello 2.1.1-4
```

) Si vous aimeriez enregistrer tous vos appels à `dpkg` (même ceux effectués en utilisant une interface telle que `aptitude`), vous pouvez ajouter

```
log /var/log/dpkg.log
```

dans votre /etc/dpkg/dpkg.cfg. Soyez sûr que votre fichier de journal sera recyclé périodiquement. Si vous utilisez logrotate, cela peut être réalisé en créant un fichier /etc/logrotate.d/dpkg qui contient

```
/var/log/dpkg {
  missingok
  notifempty
}
```

Vous trouverez plus de détails sur la journalisation de dpkg dans la page de manuel de dpkg(1).

aptitude enregistre les installations, les suppressions et les mises à jour de paquets qu'il s'apprête à effectuer dans le fichier /var/log/aptitude. Notez que les *résultats* de ces actions ne sont pas enregistrés dans ce fichier !

Une autre façon d'enregistrer vos actions est d'exécuter votre session de gestion de paquets dans le programme script(1).

9.5 Puis-je mettre à jour automatiquement le système ?

Oui. Vous pouvez utiliser cron-apt ; cet outil met à jour le système à intervalle régulier en utilisant un cron. Par défaut, il met juste à jour la liste des paquets et télécharge les nouveaux paquets mais ne les installe pas.

Note : La mise à jour automatique des paquets n'est **PAS** recommandée sur les systèmes *testing* ou *unstable*, car cela peut apporter un comportement inattendu ou supprimer des paquets sans le notifier.

9.6 J'ai plusieurs machines : comment puis-je télécharger les mises à jour seulement une fois ?

Si vous avez plus d'une machine Debian dans votre réseau, il est utile d'utiliser apt-proxy pour garder tous vos systèmes Debian à jour.

apt-proxy réduit les besoins de bande passante des miroirs Debian en diminuant la fréquence de mises à jour du fichier de paquets, publications et sources à partir de l'utilisateur et ainsi récupérer une seule fois chaque fichier, indépendamment de la requête actuelle, à partir du proxy. apt-proxy construit automatiquement un miroir HTTP Debian basé sur les requêtes qui passent par le proxy.

Pour plus de détails, consultez le site web d'apt-proxy à l'adresse http://apt-proxy.sourceforge.net/

Évidemment, vous pouvez profiter des mêmes améliorations avec un proxy standard pour lequel tous vos systèmes sont configurés.

Chapitre 10

Debian et le noyau

10.1 Puis-je installer et compiler un noyau sans spécificité Debian ?

Oui.

La bibliothèque C de Debian est construite avec la version *stable* la plus récente des entêtes du **noyau**. Si vous devez compiler un programme avec des entêtes de noyau plus récentes que celles de la branche stable, vous devriez alors mettre à jour le paquet contenant les entêtes (`libc6-dev`), ou employer les nouvelles entêtes incluses dans l'arborescence des sources du nouveau noyau installé. C'est-à-dire, si les sources du noyau sont dans `/usr/src/linux`, alors vous devriez ajouter `-I/usr/src/linux/include/` à votre ligne de commande pour la compilation.

10.2 Quel est l'outil fourni par Debian pour construire des noyaux personnalisés ?

Les utilisateurs qui souhaitent (ou doivent) construire un noyau personnalisé sont encouragés à installer le paquet `kernel-package`. Ce paquet contient le script permettant de construire le paquet du noyau et fournit la possibilité de créer un paquet `linux-image-`*`version`* simplement en exécutant la commande

```
make-kpkg --initrd kernel_image
```

dans le répertoire racine des sources du noyau. Une aide est disponible avec la commande

```
make-kpkg --help
```

et dans la page de manuel `make-kpkg(1)`.

Les utilisateurs doivent télécharger séparément les sources pour le noyau le plus récent (ou celui de leur choix) depuis leur site préféré d'archive Linux, à moins qu'un paquet `linux-source-`*`version`* soit disponible (où *version* indique la version du noyau).

Des instructions détaillées sur l'utilisation du paquet `kernel-package` sont données dans le fichier `/usr/share/doc/kernel-package/README.gz`.

10.3 Comment puis-je faire des disquettes de démarrage personnalisées ?

Cette tâche est considérablement facilitée par le paquet `boot-floppies` de Debian, que vous trouvez normalement dans la section `admin` des archives ftp de Debian. Les scripts shell de ce paquet produisent des disquettes de démarrage au format `SYSLINUX`. Ce sont des disquettes avec un formatage de type MSDOS dont le mbr (master boot records) a été modifié de sorte qu'elles puissent démarrer Linux directement (ou d'autres systèmes d'exploitation ayant été définis dans le fichier `syslinux.cfg` de la disquette). D'autres scripts de ce paquet permettent la création de disquettes de restauration et peuvent même reproduire les disques de base.

Vous trouverez plus d'informations sur ce sujet dans le fichier `/usr/doc/boot-floppies/README` après avoir installé le paquet `boot-floppies`.

10.4 Quelles dispositions spéciales Debian fournit-elle pour manipuler les modules ?

Le paquet Debian `modconf` fournit un script shell (`/usr/sbin/modconf`) qui peut être utilisé pour personnaliser la configuration des modules. Ce script présente une interface à base de menus, demandant à l'utilisateur les pilotes de périphérique qui doivent être chargés sur son système. Les réponses sont utilisées pour personnaliser les fichiers `/etc/modules.conf` (qui liste les alias et les arguments qui doivent être utilisés par les différents modules), `/etc/modutils/` et `/etc/modules` (qui contient les modules qui doivent être chargés au démarrage).

Comme les (nouveaux) fichiers `Configure.help` qui sont maintenant disponibles pour gérer la compilation de noyau personnalisé, le paquet `modconf` est fourni avec un ensemble de fichiers d'aide (dans `/usr/lib/modules_help/`) qui fournissent des informations détaillées sur les paramètres appropriés pour chaque module.

10.5 Puis-je sans risque désinstaller un vieux paquet de noyau et si oui comment ?

Oui. Le script `linux-image-`*NNN*`.prerm` vérifie si le noyau que vous utilisez actuellement est identique au noyau que vous essayez de désinstaller. Donc vous pouvez supprimer les paquets de noyau non souhaités en utilisant cette commande :

```
dpkg --purge linux-image-NNN
```

(Bien sûr, remplacez *NNN* par le numéro de version et de révision de votre noyau)

Chapitre 11

Personnaliser votre installation de Debian GNU/Linux

11.1 Comment puis-je m'assurer que tous les programmes utilisent le même format de papier ?

Installez le paquet `libpaper1`, il vous sera demandé d'indiquer le format de papier utilisé sur le système. Cette configuration sera sauvegardée dans le fichier `/etc/papersize`.

Les utilisateurs peuvent modifier la configuration du format de papier en utilisant la variable d'environnement `PAPERSIZE`. Pour plus d'informations, reportez-vous à la page de manuel `papersize(5)`.

11.2 Comment autoriser l'accès aux périphériques matériels sans compromettre la sécurité ?

La plupart des fichiers de périphérique dans le répertoire `/dev` appartiennent à des groupes prédéfinis. Par exemple, `/dev/fd0` appartient au groupe `floppy` et `/dev/dsp` au groupe `audio`.

Si vous souhaitez que certains utilisateurs puissent accéder à ces périphériques, vous devez juste ajouter les utilisateurs dans le groupe du périphérique. Faites

```
adduser utilisateur groupe
```

Cette méthode vous permet de ne pas changer les permissions sur le fichier du périphérique.

Si vous effectuez cette opération depuis l'interpréteur de commandes d'un utilisateur ou en utilisant une interface graphique, vous devez vous déconnecter puis vous reconnecter pour devenir effectivement un membre de ce groupe. Pour vérifier à quels groupes vous appartenez, lancez la commande `groups`.

Faites attention, car depuis l'introduction de `udev`, il se peut que vous modifiez les droits d'un périphérique qui seront de toute manière réglés au démarrage du système. Si cela vous arrive, vous devez ajuster les règles dans `/etc/udev`.

11.3 Comment charger une police pour la console au démarrage de Debian ?

Les paquets `kbd` et `console-tools` permettent cela. Éditez les fichiers `/etc/kbd/config` ou `/etc/console-tools/config`.

11.4 Comment configurer les paramètres par défaut des programmes X11 ?

Les programmes X de Debian installent leurs données de configuration dans le répertoire `/etc/X11/app-defaults/`. Si vous souhaitez personnaliser globalement les applications X, mettez vos personnalisations dans ces fichiers. Ils sont considérés comme fichiers de configuration, donc leur contenu sera conservé pendant les mises à jour.

11.5 Chaque distribution semble avoir une méthode de démarrage différente. Qu'en est-il de Debian ?

Comme tous les Unix, Debian exécute le programme `init` au démarrage. Le fichier de configuration pour le programme `init` (`/etc/inittab`) indique que le premier script devant être exécuté devrait être `/etc/init.d/rcS`. Ce script exécute tous les scripts présents dans `/etc/rcS.d` en incluant le source ou en « forkant » un sous-processus selon leur extension, pour exécuter les initialisations comme vérifier et monter les systèmes de fichiers, charger les modules, démarrer les services réseau, configurer l'horloge et pour exécuter d'autres initialisations. Puis, pour assurer la compatibilité, il lance aussi les fichiers (excepté ceux avec un '.' dans leur nom) de `/etc/rc.boot/`. Les scripts de ce dernier répertoire sont généralement réservés à l'administrateur système, et leur utilisation dans des paquets est obsolète.

Après avoir fini le processus de démarrage, `init` exécute tous les scripts de démarrage du niveau d'exécution (« runlevel ») par défaut (ce niveau d'exécution est indiqué par le champs `id` du fichier `/etc/inittab`). Comme la plupart des Unix compatibles System V, Linux a sept niveaux d'exécution :

— 0 (arrêt du système),
— 1 (mode mono-utilisateur ou mode maintenance),
— 2 à 5 (différents modes multi-utilisateur) et
— 6 (redémarrage du système).

Les systèmes Debian sont configurés avec id=2, ce qui implique que le niveau d'exécution par défaut est '2' lorsqu'on entre dans l'état multi-utilisateur, et ce sont les scripts de `/etc/rc2.d/` qui seront exécutés.

En fait, les scripts dans tous les répertoires `/etc/rcN.d/` sont simplement des liens symboliques vers les scripts du répertoire `/etc/init.d/`. Cependant, c'est le nom des liens dans chacun des répertoires `/etc/rcN.d/` qui indique la *manière* dont les scripts de `/etc/init.d/` seront exécutés. Spécifiquement, avant d'entrer dans un niveau d'exécution, tous les scripts commençant par un 'K' sont lancés, ces scripts arrêtent les services. Puis les scripts commençant avec 'S' sont lancés, ils démarrent les services. Le nombre à deux chiffres suivant les lettres 'K' ou 'S' indique l'ordre dans lequel seront exécutés les scripts. Les scripts ayant les chiffres les plus petits sont exécutés en premier.

Cette approche fonctionne parce que les scripts de `/etc/init.d/` prennent tous un paramètre qui peut être « start », « stop », « reload », « restart » ou « force-reload » et puis exécuteront la fonction indiquée par le paramètre. Ces scripts peuvent être aussi utilisés après le démarrage du système, pour contrôler divers services.

Par exemple, avec l'argument « reload » la commande

```
/etc/init.d/sendmail reload
```

envoie un signal au démon sendmail pour qu'il relise son fichier de configuration. À ce propos, Debian fournit `invoke-rc.d` comme programme intermédiaire (« wrapper ») destiné à appeler les scripts présents dans `/etc/init.d/`.

11.6 Il semble que Debian n'utilise pas `rc.local` pour personnaliser le processus de démarrage ; comment faire ?

Supposons que vous ayez besoin d'exécuter le script `foo` au démarrage ou lors du passage à un niveau d'exécution (System V) particulier. L'administrateur devrait :

— Mettre le script `foo` dans le répertoire `/etc/init.d/`.
— Lancer la commande Debian `update-rc.d` avec les paramètres appropriés, pour préciser quels niveaux d'exécution (« runlevels ») doivent exécuter le service, et lesquels doivent l'arrêter.
— Envisager de redémarrer le système pour vérifier que le service démarre correctement (en supposant qu'il a été configuré pour démarrer au niveau d'exécution par défaut), ou le démarrer lui-même avec la commande `/etc/init.d/foo start`.

On pourrait par exemple, exécuter le script `foo` au démarrage, en le mettant dans `/etc/init.d/` et en lançant la commande `update-rc.d foo defaults 19`. Le paramètre `defaults` se rapporte aux niveaux d'exécution, c'est-à-dire (du moins en l'absence de tout paragraphe de commentaires LSB) que le service est démarré des niveaux 2 à 5, et est arrêté aux niveaux 0, 1 et 6. Toute directive LSB Default-Start ou Default-Stop in dans le script `foo` a la priorité dans la version sysv-rc de update-rc.d, mais est ignorée dans l'actuelle (v0.8.10) version file-rc de update-rc.d. Le paramètre `19` permet de s'assurer que le script `foo` sera exécuté après la fin de l'exécution de tous les scripts avec un paramètre inférieur à 19, et avant tous les scripts avec un nombre supérieur ou égal à 20.

11.7 Comment le système de gestion de paquet traite-t-il les paquets qui contiennent des fichiers de configuration pour d'autres paquets ?

Certains utilisateurs souhaitent créer, par exemple, un nouveau serveur en installant des paquets provenant de Debian et un paquet créé localement, contenant des fichiers de configuration. Ce n'est généralement pas une bonne idée, parce que `dpkg` ne connaîtra pas ces fichiers de configuration s'ils sont dans un paquet différent, et risque de modifier les fichiers de configuration quand l'un des paquets initiaux sera mis à jour.

Au lieu de cela, créez un paquet local pour modifier les fichiers de configuration des paquets de Debian. Puis `dpkg` et le reste du système de gestion de paquet verront que les fichiers ont été modifiés par l'administrateur et n'essayeront pas de les écraser quand ces paquets sont mis à jour.

11.8 Comment remplacer un fichier installé par un paquet, de sorte qu'une version différente puisse être employée à la place ?

Supposez qu'un administrateur ou un utilisateur local souhaite utiliser un programme « login-local » plutôt que le programme « login » fourni par le paquet Debian `login`.

Ne pas faire :

— remplacer `/bin/login` avec `login-local`.

Le système de gestion des paquets ne saura rien de ce changement et écrasera simplement votre `/bin/login` personnalisé à chaque fois que `login` (ou tout autre paquet fournissant `/bin/login`) sera installé ou mis à jour.

Faites, plutôt

— Exécutez :

```
dpkg-divert --divert /bin/login.debian /bin/login
```

afin que toutes les installations futures du paquet Debian `login` écrivent le fichier `/bin/login` dans `/bin/login.debian`.

— puis tapez :

```
cp login-local /bin/login
```

pour déplacer votre programme construit localement au bon endroit.

Exécutez `dpkg-divert --list` pour obtenir la liste des remplacements actuellement actifs sur votre système.

Pour plus d'informations, lisez la page de manuel `dpkg-divert(8)`.

11.9 Comment puis-je inclure mon paquet construit localement dans la liste des paquets disponibles connus par le système de gestion des paquets ?

Lancer la commande :

```
dpkg-scanpackages BIN_DIR OVERRIDE_FILE [PATHPREFIX] > mes_Paquets
```

Où :

— BIN-DIR est le répertoire où les archives Debian (généralement ils ont « .deb » comme extension) sont conservés.
— OVERRIDE_FILE est un fichier qui est édité par les responsables de la distribution et est généralement sauvegardé sur l'archive FTP Debian dans le fichier `indices/override.main.gz` pour les paquets Debian de la section « main » Vous pouvez ignorer ce fichier pour des paquets locaux.
— PATHPREFIX est une chaîne *optionnelle* qui peut être ajoutée au début du fichier `mes_Paquets` créé.

Une fois que vous avez construit le fichier `mes_Paquets`, appelez le système de gestion des paquets en utilisant la commande :

```
dpkg --merge-avail mes_Paquets
```

Si vous utilisez APT, vous pouvez aussi ajouter votre dépôt local dans votre fichier `sources.list(5)`.

11.10 Certains utilisateurs apprécient mawk, d'autres gawk ; certains utilisent vim, d'autres elvis ; certains préfèrent trn, d'autres tin ; comment Debian gère-t-elle la diversité ?

Il y a plusieurs cas où deux paquets fournissent deux versions différentes d'un programme, où tous les deux fournissent la même fonctionnalité. Les utilisateurs pourraient préférer un plutôt qu'un autre inhabituel, ou parce que l'interface utilisateur d'un paquet est d'une façon ou d'une autre plus agréable que l'interface des autres. D'autres utilisateurs sur le même système pourraient faire des choix différents.

Debian emploie un système de paquets « virtuels » pour permettre aux administrateurs de choisir (ou laisser les utilisateurs choisir) leurs outils favoris quand il y en a plusieurs qui fournissent la même fonctionnalité de base, en répondant aux exigences de dépendance du paquet sans indiquer de paquet particulier.

Par exemple, il peut y avoir sur un système deux versions différentes d'un programme de lecture de nouvelles. L'installation d'un serveur de nouvelles peut recommander la présence d'un programme de lecture de nouvelles sur le système et laisser le choix de `tin` ou de `trn` aux utilisateurs. Ceci est possible parce que les paquets `tin` et `trn` fournissent le paquet virtuel `news-reader`. Le programme qui sera appelé est déterminé par le lien symbolique `/etc/alternatives/news-reader` pointant vers le programme choisi, par exemple `/usr/bin/trn`.

Un lien simple est insuffisant pour gérer l'utilisation complète d'un programme alternatif ; normalement, les pages de manuel, et probablement d'autres fichiers de support doivent être aussi accessibles. Le script Perl `update-alternatives` fournit le moyen de s'assurer que le système choisit bien par défaut tous les fichiers associés au paquet indiqué.

Par exemple, pour vérifier quel exécutable fournit le gestionnaire de fenêtres (x-window-manager), exécutez :

```
update-alternatives --display x-window-manager
```

Si vous souhaitez le modifier, tapez la commande :

```
update-alternatives --config x-window-manager
```

et suivez les instructions affichées à l'écran (saisissez le nombre correspondant à votre choix)

Si un paquet ne s'enregistre pas lui-même comme gestionnaire de fenêtres pour différentes raisons, (remplissez un rapport de bogue si c'est une erreur) ou si vous utilisez un gestionnaire de fenêtres présent dans le répertoire /usr/local, les choix sur l'écran ne contiendront pas votre entrée préférée. Vous pouvez mettre à jour le lien par des options de ligne de commande, comme ceci :

```
update-alternatives --install /usr/bin/x-window-manager \
  x-window-manager /usr/local/bin/wmaker-cvs 50
```

Le premier paramètre de l'option « –install » est un lien symbolique qui pointe vers /etc/alternatives/NOM, où NOM est le deuxième paramètre. Le troisième paramètre est le programme vers lequel /etc/alternatives/NOM pointe et le quatrième paramètre est la priorité (une plus grande valeur signifie que l'alternative sera très probablement sélectionnée automatiquement).

Pour supprimer une alternative que vous avez ajoutée, lancez simplement :

```
update-alternatives --remove x-window-manager /usr/local/bin/wmaker-cvs
```

Chapitre 12

Obtenir de l'aide sur Debian GNU/Linux

12.1 Quelles autres documentations existent sur le système Debian ?

— Les instructions d'installation pour la version actuelle : voir `http://www.debian.org/releases/stable/installmanual`.

— Le Guide de référence pour Debian couvre beaucoup d'aspects de l'administration du système au travers d'exemples de commande shell. Des tutoriels, des astuces et d'autres informations sont fournis pour beaucoup de sujets différents allant de l'administration système à la programmation.

Vous pouvez le trouver dans le paquet `debian-reference`, ou à l'adresse `http://www.debian.org/doc/user-manuals#quick-reference`.

— La charte Debian documente les règles pour la distribution, c.-à-d. la structure et le contenu des archives Debian, certaines questions liées à la conception du système d'exploitation, etc. Elle inclut également les impératifs techniques que chaque paquet doit satisfaire pour être inclus dans la distribution et documente les aspects techniques de base des binaires Debian et des paquets source.

Vous pouvez la trouver dans le paquet `debian-policy`, ou à l'adresse `http://www.debian.org/doc/devel-manuals#policy`.

— La documentation développée par le Projet de Documentation Debian. Elle est disponible à l'adresse `http://www.debian.org/doc/` et comprend les guides d'utilisateur, d'administration et de sécurité pour le système d'exploitation Debian GNU/Linux.

— La documentation des paquets Debian installés : la plupart des paquets ont des fichiers de documentation dans `/usr/share/doc/PAQUET`.

— La documentation sur le projet Linux. Le paquet Debian `doc-linux` installe les versions les plus récentes des guides pratiques et des mini-guides pratiques provenant du Linux Documentation Project (`http://www.tldp.org/`). (NdT : vous pourrez trouver des traductions en français de ces guides pratiques dans le paquet `doc-linux-fr-html` ou sur le site traduc.org (`http://www.traduc.org/`).)

— Les pages de manuel style Unix. La plupart des commandes ont des pages de manuel écrites dans le style des fichiers man UNIX originaux. Par exemple, pour voir la page de manuel pour la commande « ls », lancez `man ls`. Lancez `man man` pour plus d'informations sur comment trouver ou visualiser des pages de manuel.

Les nouveaux utilisateurs de Debian doivent noter que les pages de manuel de la plupart des commandes système ne sont pas disponibles tant que les paquets n'ont pas été installés.

  — `man-db` contient le programme `man` et d'autres programmes pour la manipulation des pages de manuel.

  — `manpages` contient les pages de manuel système (voir 'Comment Debian gère-t-elle les langues non anglophones ?' page 19).

— Les pages info de style GNU : La documentation d'utilisation pour beaucoup de commandes, particulièrement pour les outils GNU, est disponible non pas au format page de manuel mais dans des fichiers « info » qui peuvent être lus grâce à l'outil GNU `info`, en lançant `M-x info` depuis GNU Emacs ou avec une autre visionneuse de page d'info.

L'avantage principal par rapport aux page de manuel, c'est d'être un système hypertexte. Il ne nécessite *pas* l'utilisation d'un navigateur web, `info` peut être exécuté depuis une console texte. Il a été conçue par Richard Stallman et a précédé le web.

Remarquez que vous avez la possibilité d'accéder à beaucoup de documentation sur votre système en utilisant un navigateur web au travers des commandes dwww, dhelp ou doccentral, trouvées dans leurs paquets respectifs, ou en utilisant yelp.

12.2 Y a-t-il des ressources en ligne pour discuter de Debian ?

Oui. En fait, le moyen principal pour fournir de l'aide aux utilisateurs passe par les courriers électroniques. Nous donnerons quelques détails à ce sujet, et mentionnerons quelques autres ressources utiles. Cependant, plus de ressources sont listées sur la page du Support Debian (http://www.debian.org/support).

12.2.1 Listes de diffusion

Il y a beaucoup de listes de diffusion sur Debian (http://www.debian.org/MailingLists/).

Sur un système avec le paquet doc-debian installé, vous trouverez une liste complète des listes de diffusion dans le fichier /usr/share/doc/debian/mailing-lists.txt.

Les listes de diffusion Debian sont nommées par rapport au schéma suivant debian-*liste-sujet*. Exemples de liste : debian-announce, debian-user, debian-news. Pour s'abonner à une liste debian-*liste-sujet*, envoyez un courrier électronique à debian-*liste-sujet*-request@lists.debian.org avec pour sujet "subscribe". Assurez-vous de ne pas oublier d'ajouter *-request* à l'adresse électronique quand vous utilisez cette méthode pour vous abonner ou vous désabonner. Autrement, votre message sera envoyé sur la liste elle-même, ce qui pourrait être embarrassant ou ennuyeux, selon votre point de vue.

Vous pouvez vous abonner à une liste de diffusion en utilisant le formulaire web (http://www.debian.org/MailingLists/subscribe). Vous pouvez aussi vous désabonner en utilisant le formulaire web (http://www.debian.org/MailingLists/unsubscribe).

L'adresse électronique de l'administrateur des listes est <listmaster@lists.debian.org>, vous pouvez l'utiliser en cas de problème.

Les listes de diffusion sont des forums publics. Tous les messages envoyés sur les listes sont également copiés dans les archives publiques, afin que tout le monde (même les non-abonnés) puisse les parcourir ou chercher dedans. Veuillez vous assurer de ne jamais envoyer de contenu confidentiel ou illicite aux listes. Cela inclut des choses comme les adresses électroniques. En particulier, notez qu'il est connu que les spammers abusent des adresses électroniques postées sur nos listes de diffusion. Consultez la Charte de respect de la vie privée sur les listes de diffusion (http://www.debian.org/MailingLists/#disclaimer) pour plus informations.

Les archives des listes de diffusion Debian sont disponibles sur le web à l'adresse http://lists.debian.org/.

Quel est le code de conduite sur les listes de diffusion ?

Quand vous utilisez les listes de diffusion Debian, veuillez suivre les règles :

— N'envoyez pas de spam. voir la politique Debian concernant la publicité sur les listes de diffusion (http://www.debian.org/MailingLists/#ads).

— Ne soyez pas agressif, ce n'est pas poli. Les personnes qui aident au développement de Debian sont tous volontaires, donnant leur temps, leur énergie et leur argent afin d'essayer de faire avancer le projet Debian ensemble.

— N'employez pas un langage grossier ; certaines personnes reçoivent les listes par l'intermédiaire de paquet radio où les injures sont illégales.

— Assurez-vous d'utiliser la bonne liste. N'envoyez *jamais* vos requêtes de (dés)abonnement à la liste de diffusion [1]

— voir la section 'Comment signaler un bogue dans Debian ?' page suivante.

12.2.2 Forums web

debianHELP (http://debianhelp.org/) et Debian User Forums (http://forums.debian.net/) sont des forums web sur lesquels vous pouvez poser vos questions à propos de Debian et y obtenir des réponses de la part des autres utilisateurs (ceux-ci ne font pas officiellement partie du projet Debian).

12.2.3 Wiki

Des solutions pour des problèmes courants, des How To, des guides, des conseils et de la documentation autre peut être trouvée sur le Debian Wiki (http://wiki.debian.org/), en changements constants.

1. Utilisez l'adresse debian-*list-subject*-REQUEST@lists.debian.org pour cela.

12.2.4 Responsables

Les utilisateurs peuvent envoyer leurs questions à différents responsables de paquet en utilisant le courrier électronique. Pour contacter un responsable d'un paquet nommé xyz, envoyez un message à l'adresse *xyz@packages.debian.org*.

12.2.5 Groupes de discussion Usenet

Les utilisateurs peuvent envoyer des questions non spécifiques à Debian sur l'un des groupes de discussion Linux sur Usenet, comme comp.os.linux.* or linux.*. Il y a plusieurs listes de groupes de discussion Linux sur Usenet et d'autres ressources relatives sur le web, par exemple sur les sites Linux Online (`http://www.linux.org/docs/usenet.html`) et LinuxJournal (`http://www.linuxjournal.com/helpdesk.php`).

12.3 Y a-t-il un moyen rapide pour rechercher des informations sur Debian GNU/Linux ?

Il y a différents moteurs de recherche pour trouver de la documentation relative à Debian :

— Debian WWW search site (`http://search.debian.org/`).

— Google Groupes (`http://groups.google.com/`) : un moteur de recherche sur les groupes de discussion.

Par exemple, pour trouver quelles expériences les personnes ont avec les pilotes pour les contrôleurs Promise sous Debian, essayez de rechercher la phrase `Promise Linux driver`. Ceci vous montrera toutes les discussions qui contiennent ces chaînes, c.-à-d. celles où les gens ont discuté de ces sujets. Si vous ajoutez Debian à la chaîne de recherche, vous obtiendrez également les discussions liées spécifiquement à Debian.

— Les moteurs de recherche web, comme AltaVista (`http://www.altavista.com/`) ou Google (`http://www.google.com/`), aussi longtemps que vous employez les bons termes de recherche.

Par exemple, une recherche sur la chaîne « cgi-Perl » donne une explication plus détaillée de ce paquet que le bref champ de description du fichier control.

12.4 Y a-t-il une base des bogues connus ?

Des rapports sur des problèmes non-résolus (et fermés) sont publiquement disponibles : Debian a promis d'agir ainsi en inscrivant « Nous ne cacherons pas les problèmes » dans le contrat social Debian (`http://www.debian.org/social_contract`).

La distribution Debian GNU/Linux a un système de gestion des bogues (bug tracking system, BTS) qui classe les détails des bogues rapportés par les utilisateurs et les développeurs. Chaque bogue est numéroté et est conservé dans la base. Une fois qu'il a été traité, il est marqué comme tel.

Les copies de ces informations sont disponibles à l'adresse `http://www.debian.org/Bugs/`.

Un serveur de courrier fournit un accès au système de gestion des bogues via le courrier électronique. Afin d'obtenir les instructions, envoyez un courrier électronique à request@bugs.debian.org avec « help » dans le corps du message.

12.5 Comment signaler un bogue dans Debian ?

Si vous trouvez un bogue dans Debian, veuillez lire les instructions sur la façon de signaler un bogue dans Debian. Ces instructions peuvent être obtenues de plusieurs manières :

— Depuis le site web. Une copie des instructions se trouve sur la page `http://www.debian.org/Bugs/Reporting`.

— Sur tout système Debian ayant le paquet `doc-debian` installé. Les instructions sont dans le fichier `/usr/share/doc/debian/bug-reporting.txt`.

— Par FTP anonyme. Les sites miroir de Debian ont les instructions dans le fichier `doc/bug-reporting.txt`.

Vous pouvez utiliser le paquet `reportbug` qui vous guidera dans la rédaction du bogue et enverra le message à la bonne adresse en ajoutant automatiquement des détails concernant votre système. Il pourra aussi vous montrer les bogues déjà ouverts sur le paquet contre lequel vous rédigez votre rapport, au cas où le bogue aurait déjà été signalé et pour que vous puissiez ajouter des informations complémentaires à ce rapport de bogue.

Vous recevrez un acquittement automatique de votre rapport de bogue. le rapport de bogue sera aussi automatiquement numéroté, enregistré dans la base du BTS et expédié à la liste de diffusion debian-bugs-dist.

Chapitre 13

Participer au projet Debian

Les dons (http://www.debian.org/donations) de temps (pour développer de nouveaux paquets, maintenir les paquets existants ou fournir de l'aide aux utilisateurs), de ressources (pour les miroirs FTP et les archives WWW) et d'argent (pour payer de nouvelles plateformes de tests ainsi que du matériel pour les serveurs) peuvent aider le projet.

13.1 Comment devenir un développeur Debian ?

Le développement de Debian est ouvert à tous. Les nouveaux utilisateurs ayant les bonnes qualifications ou la motivation d'apprendre, sont nécessaires pour maintenir les paquets existants qui « ont été abandonnés » par leurs responsables précédents.

La façon de devenir un développeur Debian peut être trouvée dans Le coin des nouveaux membres Debian (http://www.debian.org/devel/join/newmaint) sur le site web de Debian.

13.2 Comment fournir des ressources au projet Debian ?

Puisque le projet vise à faire un ensemble substantiel de logiciel rapidement et facilement accessible à travers le monde, des miroirs sont un besoin urgent. Il est souhaitable mais pas absolument nécessaire de créer un miroir de toute l'archive. Veuillez consulter la page Taille d'un miroir (http://www.debian.org/mirror/size) pour obtenir des informations sur l'espace disque nécessaire.

La majeure partie du miroir peut être entièrement créée de manière automatique par des scripts, sans aucune intervention. Cependant, les problèmes ou les changements de système occasionnel peuvent exiger une intervention humaine.

Si vous avez une connexion haut débit à Internet, les ressources pour créer un miroir de tout ou d'une partie de la distribution et que vous êtes disposé à prendre le temps (ou trouver quelqu'un) pour fournir une maintenance régulière du système, alors veuillez prendre contact avec <debian-admin@lists.debian.org>.

13.3 Comment contribuer financièrement au projet Debian ?

Il est possible de faire des dons à un des deux organismes suivants qui ont une place importante dans le développement du projet de Debian.

13.3.1 Software in the Public Interest

Software in the Public Interest (SPI) est une association à but non lucratif, créée quand la FSF a retiré son patronage de Debian. Le but de l'organisation est de développer et distribuer les logiciels libres.

Nos buts sont très proches de ceux de la FSF, et nous encourageons les programmeurs à utiliser la licence « GNU General Public License » pour leurs programmes. Cependant, nous avons une vue légèrement différente du fait que nous construisons et distribuons un système Linux qui diverge dans beaucoup de détails techniques du système GNU initialement prévu par la FSF. Nous communiquons toujours avec la FSF et nous coopérons en envoyant des correctifs aux logiciels GNU et en demandant à nos utilisateurs de contribuer à la FSF et au projet GNU.

Adresse de SPI : http://www.spi-inc.org/.

13.3.2 Free Software Foundation

Il n'y a pas de relation formelle entre Debian et la Free Software Foundation. Cependant, la Free Software Foundation est responsable d'une part importante des composants logiciels de Debian, y compris le compilateur GNU C, l'éditeur GNU Emacs et une grande partie de la bibliothèque d'exécution C qui est employée par tous les programmes du système. La FSF a été pionnière dans beaucoup de ce que le logiciel libre est aujourd'hui : ils ont écrit la licence « GNU General Public License » qui est utilisée par une majorité des logiciels de Debian et ont inventé le projet GNU pour créer un système entièrement libre d'Unix. Debian devrait être considéré comme un descendant du système de GNU.

Adresse de la FSF : `http://www.fsf.org/`.

Chapitre 14

Distribuer Debian GNU/Linux dans un produit commercial

14.1 Puis-je construire et vendre des CD-ROM Debian ?

Allez-y. Vous n'avez pas besoin de permission pour distribuer quelque chose que nous avons *publié*, vous pouvez mastériser votre CD-ROM dès la fin des beta-tests. Et vous n'avez pas besoin de nous rétribuer. Naturellement, tous les fabricants de CD-ROM doivent se conformer aux licences des programmes fournis par Debian. Par exemple, beaucoup de programmes sont distribués sous la licence GPL, qui vous oblige à fournir leur code source.

En outre, nous publions une liste de fabricants de CD-ROM qui donnent de l'argent, des logiciels ou du temps au projet Debian. Nous encourageons les utilisateurs à acheter auprès de ces fabricants, les dons sont donc une bonne publicité.

14.2 Debian peut-elle être empaquetée avec des logiciels non-libres ?

Oui. Bien que tous les composants principaux de Debian soient des logiciels libres, nous fournissons un référentiel « non-free » pour les programmes qui ne sont pas librement redistribuables.

Les fabricants de CD-ROM *peuvent* distribuer les programmes que nous avons placés dans ce référentiel, selon les termes des licences ou des accords privés conclus avec les auteurs de ces logiciels. Ils peuvent aussi distribuer des logiciels non libres provenant d'autres sources sur le même CD-ROM. Cela n'est pas nouveau, des logiciels libres et des logiciels commerciaux sont maintenant distribués sur le même CD-ROM par beaucoup de fabricants. Naturellement nous encourageons toujours les auteurs de logiciel à distribuer leurs programmes en tant que logiciels libres.

14.3 Je fais une distribution Linux particulière pour « un marché vertical ». Puis-je utiliser Debian GNU/Linux comme base du système et ajouter mes propres applications au-dessus ?

Oui. Les distributions dérivées de Debian sont créées à la fois en proche collaboration avec le projet Debian lui-même et par des parties externes. On peut utiliser la structure Custom Debian Distributions (`http://cdd.alioth.debian.org/`) pour travailler ensemble avec Debian Skolelinux (`http://www.skolelinux.org/`) est un tel projet.

Il y a déjà plusieurs distributions dérivées de Debian sur le marché, comme Progeny Debian, Linspire, Knoppix et Ubuntu, qui s'adressent à un public différent de Debian GNU/Linux mais elles utilisent la plupart de nos composants dans leur produit.

Debian fournit également un mécanisme pour permettre aux développeurs et aux administrateurs système d'installer des versions locales de fichiers de telle manière que ces fichiers ne soient pas écrasés quand d'autres paquets sont mis à jour. Ceci est discuté dans la question : 'Comment remplacer un fichier installé par un paquet, de sorte qu'une version différente puisse être employée à la place ?' page 51.

14.4 Est-ce que je peux créer un paquet Debian pour mon programme commercial de sorte qu'il s'installe facilement sur n'importe quel système de Debian ?

Bien sûr. L'outil de gestion des paquets est un logiciel libre. Les paquets peuvent être libres ou non, il peut les installer tous.

Chapitre 15

Changements prévus dans la prochaine version principale de Debian

15.1 Gestion étendue pour les utilisateurs non anglophones

Debian a déjà une bonne gestion pour les utilisateurs non anglophones, voir ‘Comment Debian gère-t-elle les langues non anglophones ?’ page 19.

Nous espérons trouver des personnes qui fourniront la gestion et la traduction pour d’autres langues. Certains programmes gèrent déjà l’internationalisation, et nous sommes toujours à la recherche de traducteurs pour les catalogues de messages. Mais beaucoup de programmes restent encore à internationaliser correctement.

Le projet de traduction GNU ftp://ftp.gnu.org/pub/gnu/ABOUT-NLS travaille sur l’internationalisation des programmes GNU.

Pour Debian *Lenny*, nous travaillons à des objectifs tels que :

— internationalisation de tous les paquets utilisant debconf : les paquets utilisant le système de configuration Debian doivent permettre la traduction de tous les messages affichés à l’utilisateur lors de la configuration du paquet.

— internationalisation des descriptions des paquets : les interfaces de gestion des paquets ont été modifiées pour utiliser les traductions des descriptions.

— debian/changelog et debian/control en UTF-8. Ainsi, des noms comme ceux des contributeurs des pays asiatiques peuvent être indiqués correctement.

15.2 Démarrage plus rapide : séquence d’amorçage basée sur les dépendances

Un travail est en cours pour convertir la séquence d’amorçage de Debian (http://wiki.debian.org/LSBInitScripts/DependencyBasedBoot) afin d’utiliser un ordre dynamique et basé sur les dépendances plutôt que des numéros de séquences figés. Une fois ceci terminé, les systèmes Debian pourront démarrer bien plus vite.

15.3 Améliorations apportées à l’installateur Debian

Énormément de travail a été fait pour l’installateur Debian, dont résultent des améliorations majeures. Nous mentionnons ci-dessous deux de ces améliorations.

Il est maintenant possible de démarrer l’installateur depuis Microsoft Windows, sans avoir besoin de reconfigurer le BIOS. Lors de l’insertion d’un CD-ROM, DVD-ROM ou d’une clé USB, un programme sera automatiquement démarré, guidant pas à pas l’utilisateur pour démarrer l’installateur Debian.

L’installateur Debian prend désormais en charge de manière expérimentale les systèmes avec des périphériques ATA RAID.

15.4 Plus d'architectures

Le système complet de Debian sur d'autres architectures telles qu'ARM EABI (http://wiki.debian.org/ArmEabiPort) (nommée armel, à côté de l'ancienne arm) est susceptible d'être pris en charge pour *Lenny*. La prise en charge de SuperH (http://wiki.debian.org/SHPort) est attendue pour bientôt. Veuillez noter que même si une architecture n'est pas retenue pour la publication, il peut rester possible de l'installer et de la mettre à jour en utilisant *sid*.

15.5 Plus de noyaux

En plus de Debian GNU/Hurd, Debian est aussi portée sur les noyaux BSD, à savoir FreeBSD (http://www.debian.org/ports/kfreebsd-gnu/). Ce portage est disponible pour AMD64 (*kfreebsd-amd64*) ainsi que pour l'architecture Intel traditionnelle (*kfreebsd-i386*).

Chapitre 16

Informations générales sur la FAQ

16.1 Auteurs

La première édition de cette FAQ a été écrite et maintenue par J.H.M. Dassen (Ray) et Chuck Stickelman. Les auteurs de la réécriture de la FAQ Debian GNU/Linux sont Susan G. Kleinmann et Sven Rudolph. Après eux, la FAQ a été maintenue par Santiago Vila et plus tard par Josip Rodin. Maintenant, elle est maintenue par Javier Fernández-Sanguino Peña.

Une partie de l'information venait :

— de l'annonce de publication de la Debian-1.1, par Bruce Perens (http://www.perens.com/).
— de la FAQ Linux, par Ian Jackson (http://www.chiark.greenend.org.uk/~ijackson/).
— des archives des listes de diffusions Debian (http://lists.debian.org/),
— du manuel des développeurs dpkg et de la charte Debian (voir 'Quelles autres documentations existent sur le système Debian ?' page 53)
— de beaucoup de développeurs, volontaires, et bêtas testeurs et
— des mémoires défaillantes de leurs auteurs. :-)
— du document de Kamaraju Kusumanchi FAQ sur le choix d'une distribution Debian (http://people.cornell.edu/pages/kk288/debian_choosing_distribution.html), qui l'a gracieusement rendu GPL pour que l'on puisse l'inclure comme nouveau chapitre (voir 'Choisir une distribution de Debian' page 7)

Les auteurs souhaitent remercier tous ceux qui grâce à leur aide ont rendu ce document possible.

Toute garantie est déclinée. Toutes les marques sont la propriété de leurs propriétaires respectifs.

L'adaptation française de ce document a été réalisée par Éric Madesclair et Frédéric Bothamy.

16.2 Commentaires

Les commentaires et les ajouts de ce document sont toujours les bienvenus. Vous pouvez envoyer un courrier électronique à <doc-debian@packages.debian.org>, ou soumettre un rapport de bogue de sévérité « wishlist » contre le paquet debian-faq (http://bugs.debian.org/debian-faq).

Les commentaires sur la traduction de ce document en français devraient être envoyés à la liste debian-l10n-french (http://lists.debian.org/debian-l10n-french).

16.3 Disponibilité

La dernière version de ce document peut être lue sur le site web de Debian à l'adresse http://www.debian.org/doc/FAQ/.

Il est aussi possible de le télécharger aux formats texte, HTML, PostScript et PDF à l'adresse http://www.debian.org/doc/user-manuals#faq. Il existe aussi plusieurs traductions.

Ce document est disponible dans le paquet debian-faq, et ses traductions dans les paquets debian-faq-de, debian-faq-fr ainsi que d'autres paquets.

Les fichiers SGML originaux utilisés pour la création de ce document sont aussi disponibles dans le paquet source debian-faq ou avec SVN à : svn://svn.debian.org/svn/ddp/manuals/trunk/debian-faq et http://svn.debian.org/viewsvn/ddp/manuals/trunk/debian-faq/.

16.4 Format du document

Ce document a été écrit en utilisant la DTD SGML DebianDoc (réécrite à partir de LinuxDoc SGML). Le système SGML DebianDoc permet d'obtenir des fichiers dans de nombreux formats à partir d'une seule source, par exemple ce document peut être visualisé en HTML, texte, TeX DVI, PostScript, PDF, ou GNU info.

Les utilitaires de conversion pour DebianDoc SGML sont disponibles dans le paquet Debian debiandoc-sgml.